



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA ELEKTROTECHNIKY  
A KOMUNIKAČNÍCH TECHNOLOGIÍ**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

**ÚSTAV MIKROELEKTRONIKY**

DEPARTMENT OF MICROELECTRONICS

**LABORATORNÍ ÚLOHA MĚŘENÍ VELIČIN POMOCÍ  
POLOHOVÉHO MEMS SENZORU**

LABORAL TASK OF VARIABLES MEASUREMENT BY POSITION MEMS SENSOR

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

**Jakub Jarůšek**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**doc. Ing. Jaromír Hubálek, Ph.D.**

**BRNO 2020**

# Bakalářská práce

bakalářský studijní program **Mikroelektronika a technologie**

Ústav mikroelektroniky

**Student:** Jakub Jarůšek

**ID:** 203238

**Ročník:** 3

**Akademický rok:** 2019/20

## NÁZEV TÉMATU:

### Laboratorní úloha měření veličin pomocí polohového MEMS senzoru

## POKYNY PRO VYPRACOVÁNÍ:

Práce je zaměřena na vytvoření nové měřicí úlohy do předmětu Mikrosenzory a mikroelektromechanické systémy. Cílem práce je vytvořit přípravek a obvodové řešení včetně zadání a popisu úlohy pro měření vibrací, úhlu náklonu, příp. dalších. Bakalářská práce se bude skládat z těchto dílčích úkolů:

- 1) nastudovat problematiku jednoosého akcelerometru a gyroskopu realizovaného v technologii MEMS (např. od společnosti Analog Devices),
- 2) navrhnete kompletní systémové uspořádání úlohy,
- 3) vypracujete zadání a popis úlohy,
- 4) navrhnete přípravek v podobě aktuátoru, kterým bude možné měnit měřenou veličinu, a na který bude umístěn MEMS akcelerometr/gyroskop,
- 5) navrhnete vhodný elektronický obvod zajišťující čtení a zobrazení dat,
- 6) realizujete celé řešení a ověřte jeho funkci.

## DOPORUČENÁ LITERATURA:

Podle doporučení vedoucího práce.

**Termín zadání:** 3.2.2020

**Termín odevzdání:** 8.6.2020

**Vedoucí práce:** doc. Ing. Jaromír Hubálek, Ph.D.

**doc. Ing. Jiří Háze, Ph.D.**  
předseda rady studijního programu

## UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

## **ABSTRAKT**

Technologie MEMS a její produkty jsou v dnešní době velice využívány. Jedním druhem těchto produktů jsou akcelerometry, které hrají velkou roli při určování orientace nebo kvality opotřebení, popřípadě v seismologii. Jejich funkcí se využívá jak pro spotřební elektroniku, tak i ve sférách vývoje a diagnostiky. Pro náročnější aplikace, zejména určení orientace se používají produkty zvané gyroskopy (samotné nebo ve spolupráci s akcelerometry).

## **KLÍČOVÁ SLOVA**

MEMS, senzor, akcelerometr, náklon a rotace, vibrace, ADXL.

## **ABSTRACT**

MEMS technology and her products are very used today. One of these products are accelerometers, which are highly used for determining orientation, wear measurement or seismology. Their functions are used for consumer electronics, even for development or diagnostics. For more demanding applications, especially for determining orientation, are used components named gyroscopes (alone or with cooperation with accelerometers).

## **KEYWORDS**

MEMS, sensor, accelerometer, pitch and rotation, vibration, ADXL.

JARŮŠEK, J. *Laboratorní úloha měření veličin pomocí polohového MEMS senzoru*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav mikroelektroniky, 2020. 53 s., 8 s. příloh. Bakalářská práce. Vedoucí práce: doc. Ing. Jaromír Hubálek, Ph.D.

# Prohlášení autora o původnosti díla

**Jméno a příjmení studenta:** *Jakub Jarišek*

**VUT ID studenta:** *203238*

**Typ práce:** *Bakalářská práce*

**Akademický rok:** *2019/20*

**Téma závěrečné práce:** *Laboratorní úloha měření veličin pomocí polohového MEMS senzoru*

*Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce. Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.*

V Brně dne: **8. června 2020**

.....  
Podpis autora

# PODĚKOVÁNÍ

Děkuji vedoucímu mé bakalářské práce doc. Ing. Jaromírovi Hubálkovi, Ph.D za poskytnutí přístupu na stránky Web of Science a nastínění možných měření pro tuto úlohu. Dále děkuji panu Ing. Jiřímu Veselému za obstarání potřebných součástek. Také děkuji Michalovi Hráčkovi za vytisknutí potřebných částí aktuátoru na 3D tiskárně. Dále chci poděkovat Ing. Alexandrovi Otáhalovi a Pavlovi Jánešovi za pomoc při zapájení pouzder LGA a LCC. V poslední řadě bych chtěl poděkovat své rodině za morální podporu při zpracování bakalářské práce.

# OBSAH

<b>Úvod</b>	<b>1</b>
<b>1 Mems</b>	<b>2</b>
1.1 Základy .....	2
1.2 MEMS transduktor .....	2
1.3 Mechanické snímače .....	2
1.3.1 Piezorezistivní snímač .....	3
1.3.2 Piezoelektrický snímač .....	3
1.3.3 Rezonanční snímač .....	4
1.3.4 Kapacitní snímač .....	4
<b>2 Akcelerometry a gyroskopy</b>	<b>5</b>
2.1 Akcelerometr .....	5
2.2 Gyroskop .....	6
<b>3 Popis součástek</b>	<b>8</b>
3.1 Akcelerometr ADXL345 .....	8
3.2 Akcelerometr ADXL203 .....	9
3.3 A/D převodník MCP3202 .....	10
3.4 Raspberry Pi .....	12
3.4.1 Raspberry Pi 3 model B .....	12
3.4.2 Nastavení mikropočítače raspberry Pi 3b .....	13
<b>4 Náklon</b>	<b>16</b>
4.1 Teorie výpočtu náklonu a rotace .....	16
4.2 Zapojení .....	18
4.2.1 Deska .....	18
4.2.2 Propojení akcelerometru ADXL345 a mikropočítače Raspberry Pi ..	19
4.2.3 Systém změny náklonu a rotace .....	20
4.3 Popis použitých knihoven .....	22
4.3.1 SMBUS .....	22
4.3.2 Tkinter .....	23
4.3.3 Math .....	24
4.3.4 Time .....	24

4.4	Vytvořené funkce pro sepsání programu .....	24
4.4.1	Postup pro studenty k sestrojení programu .....	26
<b>5</b>	<b>Vibrace</b>	<b>28</b>
5.1	Teorie k problematice vibrací .....	28
5.2	Popis zapojení .....	29
5.3	Zvolené knihovny .....	32
5.3.1	Spidev .....	32
5.3.2	Pyqtgraph .....	33
5.3.3	Numpy a Scipy .....	34
5.4	Princip měření .....	34
5.5	Rychlá Fourierova transformace – FFT .....	35
<b>6</b>	<b>Měření</b>	<b>37</b>
6.1	Měření s akcelerometrem ADXL345 .....	37
6.2	Měření s akcelerometrem ADXL203 .....	43
<b>7</b>	<b>Zadání</b>	<b>47</b>
<b>8</b>	<b>Závěr</b>	<b>48</b>
	<b>Literatura</b>	<b>49</b>
	<b>Seznam symbolů, veličin a zkratk</b>	<b>52</b>
	<b>Seznam obrázků</b>	<b>52</b>
	<b>Seznam tabulek</b>	<b>53</b>
<b>A</b>	<b>Návrh desky s akcelerometrem ADXL345</b>	<b>54</b>
A.1	Deska plošného spoje s ADXL345 – Top (strana součástek) .....	54
A.2	Deska plošného spoje s ADXL345 – rozmístění součástek .....	55
A.3	Deska plošného spoje s ADXL345 – Bottom (zrcadleno) .....	55
<b>B</b>	<b>návrh desky s akcelerometrem ADXL203</b>	<b>56</b>
B.1	Deska plošného spoje s ADXL203 – Top (strana součástek) .....	56
B.2	Deska plošného spoje s ADXL203 – rozmístění součástek .....	56
B.3	Deska plošného spoje s ADXL203 – Bottom (zrcadleno) .....	57
B.4	Schéma zapojení desky s ADXL203 .....	57
<b>C</b>	<b>návrh částí aktuátoru pro změnu náklonu a rotace</b>	<b>59</b>



C.1	Návrh základny aktuátoru (měřítko M1:5) .....	59
C.2	Návrh držáku pro rotační stolek 2 (měřítko M1:1).....	60
C.3	Návrh držáku pro desku s akcelerometrem ADXL345 (měřítko M1:1).	61

# ÚVOD

Technologie MEMS neboli mikro-elektromechanické systémy hrají velikou roli v elektronice, především v senzorce. MEMS se mohou nacházet v různých výrobních a výzkumných procesech počínaje od mechanických transduktorů, přes transduktory pracující s radiačními, teplotními nebo magnetickými silami, až po systémy, pracující s chemickým a biologickým materiálem a s malým množstvím vody (mikrofluidní systémy). Při měření zrychlení se používají snímače nazývané akcelerometry. Tyto součástky umožňují změřit statické vlastnosti zrychlení vůči gravitační síle Země nebo dynamické vlastnosti zrychlení, které vznikají pohybem snímače.

Cílem práce je vytvořit laboratorní úlohu pro studenty předmětu Mikrosenzory a mikroelektromechanické systémy. Vytvořená úloha má seznámit studenty s problematikou MEMS akcelerometrů. Úloha se skládá ze dvou částí. V první části se měří statické zrychlení v závislosti na působení gravitačního zrychlení Země pomocí akcelerometru ADXL345. Druhá část obsahuje měření vibrací pomocí akcelerometru ADXL203, které představují dynamicky se měnící signál.

Oba akcelerometry komunikují s okolím pomocí mikropočítače Raspberry Pi 3b, ke kterému jsou připojeny pomocí sériové komunikace. Pro akcelerometr ADXL345 je vytvořen aktuátor pro změnu náklonu a rotace. Náklon a rotace zajišťuje změnu působení gravitačního zrychlení na osy akcelerometru. V této části studenti naprogramují komunikaci a zobrazení dat pomocí funkcí z předpřipravené knihovny jazyka Python. V druhé části úlohy jsou změřeny vibrace, které se vytvoří otáčkami lopatek ventilátoru. Na rám ventilátoru je připevněn akcelerometr ADXL203 zajišťující měření. Zde je vytvořen program pro zobrazení měřených vibrací v časovém průběhu a ve frekvenčním spektru.

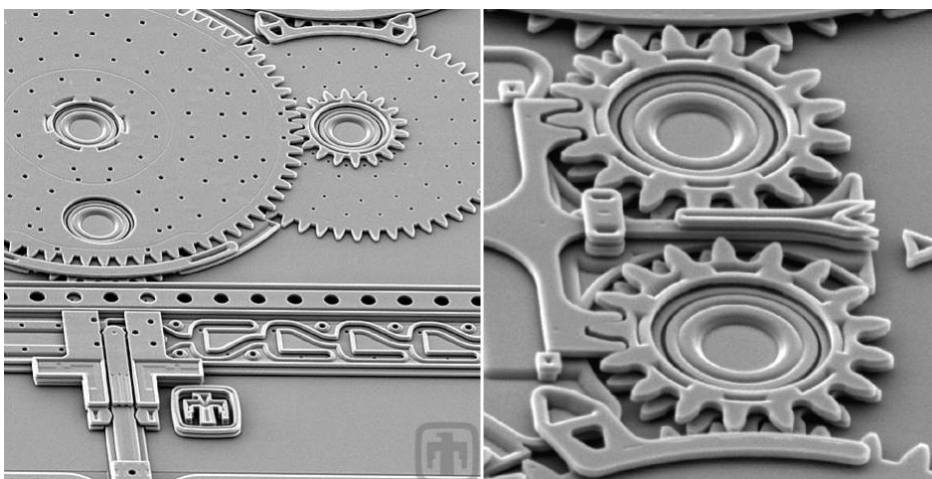
Text je rozdělen do čtyř základních částí. Kapitoly 1 a 2 zajišťují seznámení s technologií MEMS a popis funkce akcelerometrů a gyroskopu. Kapitola 3 zajišťuje seznámení s použitými akcelerometry, A/D převodníkem a mikropočítačem Raspberry Pi 3b. Kapitola popisuje jejich parametry, funkce a principy operace. Kapitoly 4 a 5 popisují problematiku obou částí úlohy a obsahují řešení dané problematiky jak z hardwarové, tak i softwarové části. Kapitola 6 obsahuje samotné měření úlohy, jednoduchý postup, a srovnání teoretických a naměřených hodnot.

# 1 MEMS

## 1.1 Základy

Mikro-elektromechanické systémy (MEMS) je technologie, používaná pro vytváření malých integrovaných zařízení nebo systémů, které spojují použití mechanických a elektrických součástí. Ty jsou vyrobeny pomocí integrovaného obvodu a mohou dosahovat velikostí od několika mikrometrů až po milimetry. Tato zařízení poté mají schopnost snímat, kontrolovat a ovládat určité fyzikální a chemické děje.

Zatímco elektronika pro tyto zařízení je vytvářena standartní technologií integrovaných obvodů, mechanické části jsou vytvářeny sofistikovanou manipulací křemíku a dalších materiálů pomocí mikroobráběcích (micromachining) procesů. Tyto procesy odstraňují jednotlivé části křemíku nebo přidávají další struktury, aby vytvořily mechanické a elektromechanické části. Pomocí těchto procesů jdou pak vytvořit struktury, jako jsou páčky, ozubená kola, písty a motory a jiné (obrázek 1). [1]



Obrázek 1 Vytvořené MEMS struktury pomocí mikroobráběcích procesů [2]

## 1.2 MEMS transduktor

Mikrosenzory a mikroaktuátory jsou základem mikro-elektromechanického systému. Mikrosenzor zjistí změnu v daném prostředí, kterou měří. Tuto změnu vysílá na výstupu jako elektrický signál. Signál jde na zpracování do nějaké logické jednotky (procesor, počítač atd.). Jednotka signál zpracuje a na základě zpracování vyšle signál do mikroaktuátoru, který vyvolá změnu úměrnou prostředí. [1]

## 1.3 Mechanické snímače

Sem spadají různé druhy zařízení, které slouží k zaznamenání vlivu nebo změny okolí, jenž byla vyvolána změnou nebo působením mechanické veličiny (síla, poloha, zrychlení,

energie a jiné) na snímač. Liší se od sebe principem, kterým senzory požadovanou veličinu měří. [1]

### 1.3.1 Piezorezistivní snímač

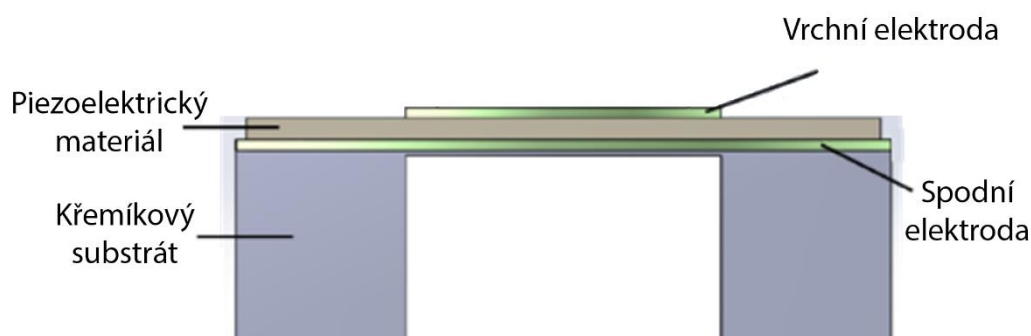
Pracuje na principu piezorezistivního jevu. Pomocí piezorezistivního materiálu nebo piezorezistivního povlaku na pohyblivé struktuře (membráně) se vlivem okolních podmínek mění rozměry dané vrstvy. Důsledkem je změna odporu ve snímači (obrázek 2). Změna odporu lze poté převést na elektrický signál o určité hodnotě. Samotný jev je mnohem výraznější u polovodičů než u kovů, což dělá z křemíku velmi citlivý snímač tlaku. [2]



Obrázek 2 Princip piezorezistivní MEMS snímače [3]

### 1.3.2 Piezoelektrický snímač

Působením vnější jevů se mechanicky deformuje piezoelektrický krystal. Tato deformace způsobí změnu potenciálu samotného krystalu, kterou je možné sledovat na elektrodách (obrázek 3). Změna potenciálu je úměrná působící síle. Tato závislost pracuje oboustranně, ovšem i při použití velkého napětí je deformace krystalu malá. Samotný křemík není piezoelektrická látka, proto se musí nanést tenká vrstva z vhodného materiálu. [3]



Obrázek 3 Princip piezoelektrického MEMS snímače [3]

### 1.3.3 Rezonanční snímač

Je tvořen mikropaprsky nebo mikromosty, které jsou nastaveny, aby oscillovaly na jejich rezonanční frekvenci. Tyto paprsky (mosty) jsou připojeny k membráně. Pohyb membrány, způsobené změnami prostředí, mění rezonanční frekvenci. Změna rezonanční frekvence je úměrná měřené veličině. [1]

### 1.3.4 Kapacitní snímač

Kapacitní měření je jedno z nejdůležitějších a nejvíce využívaných precizních měření. Mechanismus se skládá z jedné (nebo více) upevněné elektrody a jedné (nebo více) pohyblivé elektrody. Elektrody se pak chovají jako kapacitor. Pomocí pohyblivé elektrody se při působení okolních jevů mění parametry rovnice 1. Měření malých změn parametrů kapacitoru a jejich přesnost zajišťuje velmi přesné snímání senzoru. [5]

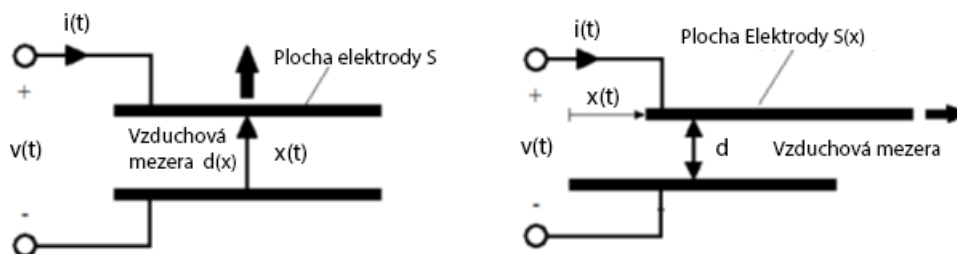
$$C = \varepsilon_0 \varepsilon_r \frac{S}{d}, \quad (1)$$

kde konstanta  $\varepsilon_0$  je permitivita vakua ( $8,854 \cdot 10^{-12} \text{ F} \cdot \text{m}^{-1}$ ),  $\varepsilon_r$  je relativní permitivita, která závisí na použitém materiálu mezi elektrodami (pro vzduch je rovna 1),  $S$  je aktivní plocha elektrod kondenzátoru a  $d$  je vzdálenost mezi elektrodami. [1]

Pro výrobu kapacitních snímačů se používají dva principy, které jsou založeny na změně rozdílných parametrů kapacitoru.

První princip je pevná plocha, proměnná mezera: nejvíce využívaný princip. Při působení vnější síly kolmo na elektrody se pohyblivá elektroda přibližuje (oddaluje). Tento pohyb zapříčiní změnu velikosti vzduchové mezery a tím i změnu kapacity, obrázek 4 vlevo. Malé změny mezery mohou vyvolat velké změny kapacity, díky čemu je tato metoda velice přesná. Nevýhodou tohoto principu je jeho nelinearita vyplývající z rovnice

Druhý princip je pevná mezera, proměnná plocha: vhodné pro detekce větších amplitud. Vlivem okolních sil působících rovnoběžně s elektrodami se pohyblivá elektroda vychyluje a vzájemná plocha elektrod se mění, jak je vidět na obrázku 4 vpravo. Kvůli přímé závislosti na ploše je změna kapacity oproti principu s proměnnou mezerou lineární. Nevýhodou je, že tento princip méně přesný. [6]



Obrázek 4 Princip kapacitního senzoru s proměnnou mezerou (vlevo) a proměnnou plochou (vpravo) [5]

## 2 AKCELEROMETRY A GYROSKOPY

### 2.1 Akcelerometr

Akcelerometr je elektromechanické zařízení určené pro měření zrychlení  $a$  [ $\text{m}\cdot\text{s}^{-2}$ ]. Zrychlení může být statické, například konstantní síla působící gravitace nebo dynamické, způsobené pohybem nebo vibracemi. Využití akcelerometrů je tak velice rozšířené, a to od určování náklonu, až po diagnostiku opotřeбенí zařízení a měření seismických jevů. [5]

V principu pracuje akcelerometr jako těleso upevněné na pružině. Při působení vnější síly na těleso vzniká podle, druhého Newtonova pohybového zákona, zrychlení:

$$\vec{F} = m * \vec{a}, \quad (2)$$

kde  $\vec{F}$  je vektor síly působící na těleso,  $m$  je hmotnost tělesa a  $\vec{a}$  je vektor zrychlení tělesa po působení síly (akcelerace). [7]

Při určení osy pohybu (například  $x$ ) pak lze dostat rovnici již pro skalární veličiny:

$$F_x = m * a_x, \quad (3)$$

kde  $F_x$  je velikost síly působící ve směru osy  $x$ ,  $m$  je hmotnost tělesa a  $a_x$  je velikost zrychlení ve směru osy  $x$ .

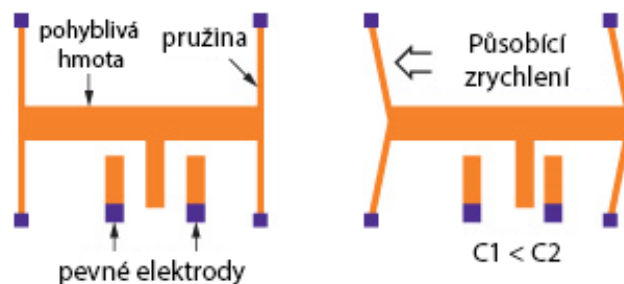
Z pohybové rovnice pružinového oscilátoru vyplývá, že vzniklé zrychlení se může zjistit pomocí velikosti vychýlení tělesa:

$$\vec{F} = m * \vec{a} = k * \vec{y}, \quad (3)$$

kde  $k$  je tuhost pružiny a  $\vec{y}$  je vektor vychýlení, způsobené působící silou.

Toto provedení je implementováno do snímače zrychlení, který může využívat různé druhy elektrických a mechanických jevů. Mezi nejčastější druhy patří kapacitní, piezoelektrické a piezorezistivní snímače.

Pro kapacitní akcelerometr je změna zrychlení vyjádřena jako změna kapacity mezi pevnými a pohyblivými elektrodami (proměnná mezera nebo proměnná plocha), obrázek 5. Změna kapacity vyvolá změnu napětí podle rovnice 1, pomocí něhož lze určit zrychlení působící na senzor. Díky své velké citlivosti jsou vhodné pro přesná měření a jsou vhodné pro měření statických jevů. [5]



Obrázek 5 Princip kapacitního akcelerometru s proměnou mezerou [8]

Změna zrychlení u piezoelektrických akcelerometrů závisí na míře mechanické deformace krystalu, která v důsledku vytvoří určité napětí. Vzniklé napětí je poté interpretováno jako zrychlení působící na snímač. Mezi výhody piezoelektrických akcelerometrů patří široké frekvenční pásmo, linearita a velmi dobré dynamické vlastnosti, kvůli kterým se hodí například pro měření vibrací. [9]

Piezorezistivní akcelerometr využívá změny odporu při mechanickém namáhání piezorezistivních materiálů. Vhodný pro vysoké frekvence a detekce šoků. Využívají se v automobilovém průmyslu pro bezpečnostní systémy. [10]

Akcelerometry se dělí do dvou základních skupin, a to analogové a digitální. Výstupní veličina analogových akcelerometrů bývá zpravidla napětí (u rezonančních i frekvence), které je závislé na změně parametrů při působení vnější síly (záleží na druhu). Součástka pak obsahuje samotný snímač a zesilovače pro zjištění hodnoty zrychlení (v podobě napětí) měřicími přístroji. Výstup digitálních akcelerometrů je číslo v bitovém (digitálním) tvaru, který udává, jak velké je vzniklé zrychlení oproti gravitačnímu zrychlení způsobené gravitací Země ( $g = 9,806 \text{ m} \cdot \text{s}^{-2}$ ). Zrychlení 2 g tak představuje dvojnásobně větší zrychlení, než je zrychlení způsobené gravitací ( $2 g = 19,612 \text{ m} \cdot \text{s}^{-2}$ ). Taková součástka pak musí obsahovat místo samotného snímače a zesilovače také A/D převodník a kontrolní logickou jednotku, která zajišťuje správnou funkci akcelerometru a komunikaci s okolím.

Další dělení akcelerometrů může být podle počtu os, po kterých měří zrychlení. Jedná se tak o jednoosé (zpravidla osa X), dvojosé (osa X a Y) a tříosé (osa X, Y a Z).

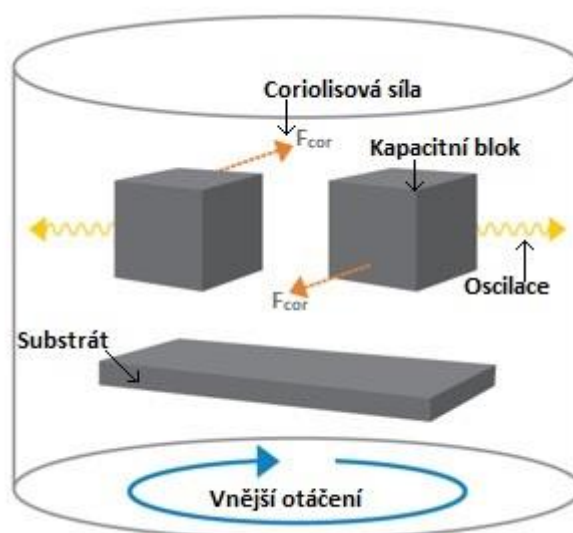
## 2.2 Gyroskop

Gyroskop je zařízení určené pro měření nebo udržení své stávající polohy a úhlové rychlosti. Skládá se z rotoru, který je umístěn na ose otáčení. Pro osu otáčení je zajištěn volný pohyb. Tento pohyb zajišťuje umístění osy do kardanového závěsu, což jsou tři kruhové prstence (gimbal) s ložisky s minimální hodnotou tření, aby izolovaly rotor. Kardanový závěs poskytuje třístupňovou volnost gyroskopu

Působením vnější rotační síly klade točící se rotor odpor na změnu jeho rotace. Tento jev se nazývá precese. Změna rotace je detekována a pomocí jiného zařízení (například motoru) je aplikována v opačném směru. Tím se vyruší precese a osa otáčení nezmění svoji orientaci. [11]

Při určení orientace může gyroskop měřit tři typy úhlového posunu:  
vychýlení – horizontální rotace na rovném povrchu, při pohledu ze shora  
náklon – vertikální rotace při pohledu zepředu  
rotace – vertikální rotace při pohledu z boku [12]

V MEMS provedení se využívá spojení mechanických oscilací a Coriolisové síly. Tato síla působí na objekt, který se pohybuje po otáčejícím předmětu. I přesto, že se objekt pohybuje přímočaře, se vlivem působení této síly vychýlí jeho trajektorie od požadované. [13] Princip MEMS gyroskopu je zobrazen na obrázku 6. Kapacitní blok uvnitř MEMS gyroskopu osciluje na pevně dané frekvenci opačnou fází než druhý blok. Vlivem otáčení, díky Coriolisové síle, se bloky odtahují. Tato změna pole změní kapacitu vůči substrátu, který se nachází pod kapacitními bloky a je úměrná změně polohy. [14]



Obrázek 6 Princip MEMS gyroskopu [14]



## 3 POPIS SOUČÁSTEK

### 3.1 Akcelerometr ADXL345

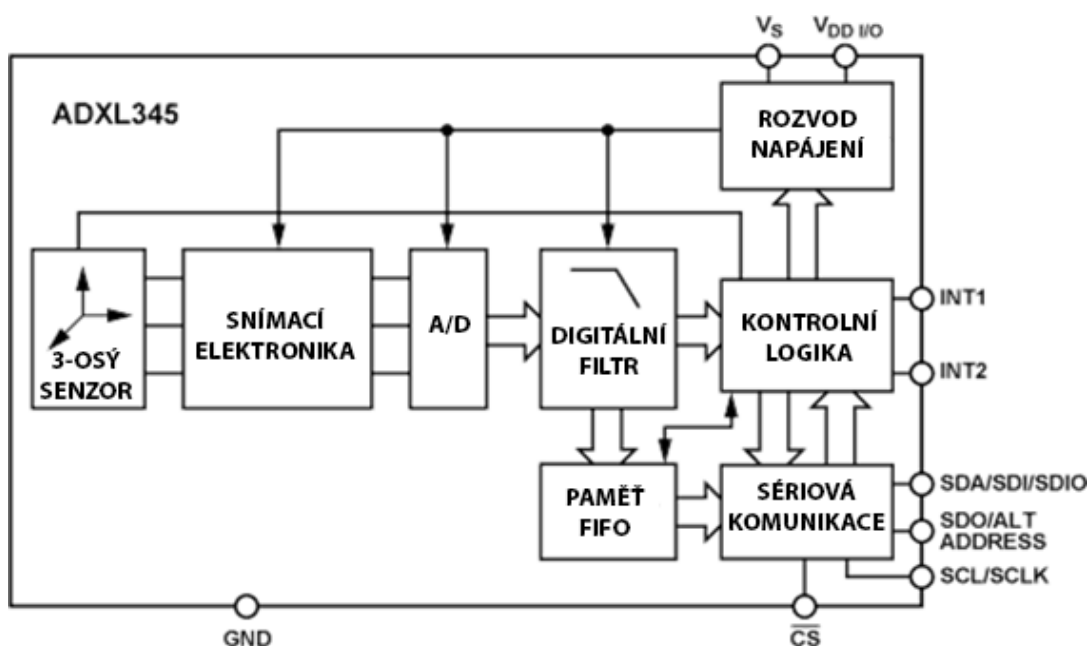
Akcelerometr ADXL345 je tříosý akcelerometr od firmy Analog Devices, který dokáže měřit zrychlení v nestavitelném rozsahu, a to  $\pm 2 \text{ g}/4 \text{ g}/8 \text{ g}/16 \text{ g}$ . Kontrolní logika pracuje s osmibitovými registry. Výčet registrů je vidět na obrázku 8. Digitální výstup pro jednu osu je tvořen dvěma osmibitovými registry a jeho hodnota může být desetibitová nebo třináctibitová (záleží na zvolené přesnosti). Všechny změřené hodnoty mohou být uloženy do vnitřní paměti o velikosti 32 vzorků, kde jeden vzorek je 16 bitů, typu FIFO. Tato paměť slouží pro ulehčení procesní paměti řídicího kontroléru (mikroprocesor, mikropočítač).

Adresa		Název	Typ	Hodnota po resetu	Popis
Hex	Dec				
0x00	0	DEVID	R	11100101	Identifikační číslo
0x01 to 0x1C	1 to 28	Reserved			Rezervované, nepoužívat
0x1D	29	THRESH_TAP	R/W	00000000	Reference šoku
0x1E	30	OFSX	R/W	00000000	Offset osy X
0x1F	31	OFSY	R/W	00000000	Offset osy Y
0x20	32	OFSZ	R/W	00000000	Offset osy Z
0x21	33	DUR	R/W	00000000	Délka šoku
0x22	34	Latent	R/W	00000000	Doba odstupu šoků
0x23	35	Window	R/W	00000000	Okno šoků
0x24	36	THRESH_ACT	R/W	00000000	Práh aktivního režimu
0x25	37	THRESH_INACT	R/W	00000000	Práh neaktivního režimu
0x26	38	TIME_INACT	R/W	00000000	Doba zrychlení pro přepnutí do neaktivního režimu
0x27	39	ACT_INACT_CTL	R/W	00000000	Povolení os pro aktivní a neaktivní režim
0x28	40	THRESH_FF	R/W	00000000	Práh zrychlení volného pádu
0x29	41	TIME_FF	R/W	00000000	Doba volného pádu
0x2A	42	TAP_AXES	R/W	00000000	Povolení os pro jeden nebo dva šoky
0x2B	43	ACT_TAP_STATUS	R	00000000	Nastavení zdrojů pro šoky
0x2C	44	BW_RATE	R/W	00001010	Rychlost čtení dat a napěťový mód
0x2D	45	POWER_CTL	R/W	00000000	Funkce šetření spotřeby
0x2E	46	INT_ENABLE	R/W	00000000	Povolení přerušení
0x2F	47	INT_MAP	R/W	00000000	Nastavení výstupu přerušení
0x30	48	INT_SOURCE	R	00000010	Zdroj přerušení
0x31	49	DATA_FORMAT	R/W	00000000	Formát dat
0x32	50	DATA0	R	00000000	Osa X, data 0
0x33	51	DATA1	R	00000000	Osa X, data 1
0x34	52	DATA0	R	00000000	Osa Y, data 0
0x35	53	DATA1	R	00000000	Osa Y, data 1
0x36	54	DATA0	R	00000000	Osa Z, data 0
0x37	55	DATA1	R	00000000	Osa Z, data 1
0x38	56	FIFO_CTL	R/W	00000000	Nastavení paměti FIFO
0x39	57	FIFO_STATUS	R	00000000	Stav paměti FIFO

Obrázek 7 Popis a adresy digitálního akcelerometru ADXL345 [15]

Akcelerometr dokáže měřit jak statistické zrychlení způsobené gravitací, tak i dynamické zrychlení způsobené pohybem nebo šokem. Mezi jeho funkce patří detekce zrychlení v porovnání s nastavenou referencí (threshold), detekce volného pádu a detekce jednoho nebo dvou šoků. Nabízí také režim nízkého napájení (low power mode), který při určité velikosti zrychlení dokáže aktivovat akcelerometr zpět do měřícího režimu.

Blokové schéma akcelerometru ADXL345 je zobrazeno na obrázku 8. Základem součástky je tříosý senzor (snímač), vytvořen pomocí kapacitních MEMS senzorů. Zjištěné napětí (analogový signál) projde příslušným obvodem a pomocí A/D převodníku je převeden na digitální signál. Tento signál je vyfiltrován, aby nevznikly chyby ve snímání. Vyfiltrované digitální hodnoty jdou do kontrolní jednotky, která je buď uloží do interní FIFO paměti nebo pošle pomocí sériového výstupu pryč. Celkový přísun energie zajišťuje blok řízení napájení. Piny  $V_s$  a  $V_{DD\ I/O}$  slouží společně s pinem GND k přísunu napětí na čip, kde  $V_s$  je celkový zdroj napětí a  $V_{DD\ I/O}$  slouží jako zdroj napětí pro digitální část a GND značí zem akcelerometru. INT1 a INT2 souží jako výstupy pro přerušení, vzniklé pomocí určité funkce, kterou akcelerometr nabízí. SDA/SDI/SDIO a SDO/ALT ADDRESS slouží pro komunikaci po sériové lince a jejich konečné určení závisí na použitém protokolu (SPI/ I<sup>2</sup>C).  $\overline{CS}$  slouží pro výběr čipu (chip select) a zvolí se nastavením na napěťovou úroveň  $V_{DD\ I/O}$  (logická 1). [15]



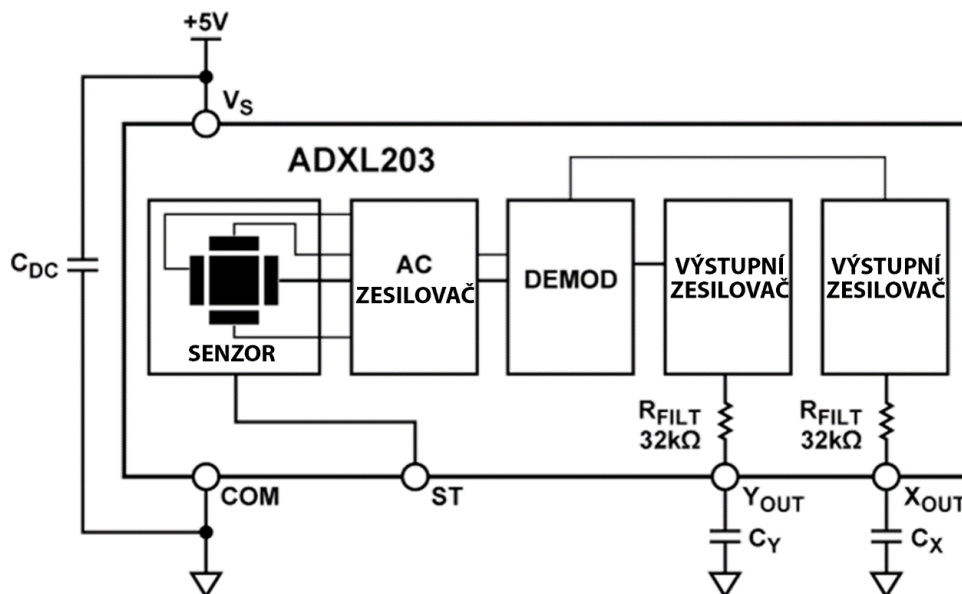
Obrázek 8 Vnitřní blokové schéma digitálního akcelerometru ADXL345 [15]

## 3.2 Akcelerometr ADXL203

ADXL203 je přesný akcelerometr s nízkou spotřebou, zajišťující měření zrychlení ve dvou osách. Jedná se tak o dvouosý, kapacitní akcelerometr, jehož osy bývají z pravidla označovány jako X a Y. Výstup ADXL203 je analogový v podobě napětí. Rozsah měření zrychlení v jednotkách gravitačního zrychlení je  $\pm 1,7$  g. Akcelerometr může měřit jak dynamické zrychlení (vibrace), tak i zrychlení statické (působení gravitace). Kvůli svému nízkému šumu může měřit signál pod hodnotou 1 mg. Pomocí přídavných kondenzátorů ( $C_x$ ,  $C_y$ ) může být vybráno potřebné měřicí frekvenční pásmo (od 0,5 Hz po 2,5 kHz). Toto frekvenční pásmo může být nastaveno pro každou osu zvlášť.

Na obrázku 9 je zobrazeno blokové schéma akcelerometru ADXL 203. Pomocí dvouosého senzoru, který pracuje na principu kapacitního snímače, je změřené zrychlení

v podobě obdélníkového signálu napětí. Hodnota amplitudy tohoto signálu je úměrná změřenému zrychlení. Pomocí analogových zesilovačů je výstupní napětí zesíleno. Zesílené napětí vstupuje do fázově citlivého demodulátoru, který signál upraví, a určí směr zrychlení. Demodulovaný signál je zesílen výstupním zesilovačem a přes vnitřní rezistor  $32\text{ k}\Omega$  je analogový signál vyveden ven (pin  $X_{\text{out}}$ ,  $Y_{\text{out}}$ ). Na svorky  $V_s$  a COM je připojené napájecí napětí, které zároveň určuje senzitivitu měření (pro  $5\text{ V}$  platí  $1\text{ V/g}$ , pro  $3\text{ V}$  platí  $560\text{ mV/g}$ ). [16]



Obrázek 9 Vnitřní blokové schéma analogového akcelerometru ADXL203 [16]

### 3.3 A/D převodník MCP3202

A/D převodník MCP3202 je převodník, který snímá analogovou hodnotu pomocí metody postupné aproximace. Tento převodník poskytuje dvanáctibitové rozlišení snímajícího signálu a pracuje při napájecím napětí  $2,7\text{ V}$  až  $5,5\text{ V}$ . MCP3202 je programovatelný a jeho vstupy poté mohou být brány jako jeden pseudo-rozdílový vstup nebo dva samostatné vstupy. Komunikace s A/D převodníkem je uskutečněna pomocí jednoduché sériové komunikace pracující pomocí protokolu SPI (módy 0,0 a 1,1). Rychlost snímání vzorku je závislá na přiloženém napájecím napětí a dosahuje rychlosti od  $50\text{ kps}$  při napětí  $2,7\text{ V}$ , až po rychlost  $100\text{ kps}$  při napětí  $5\text{ V}$ . [17]

Pro určení digitální hodnoty převedeného analogového napětí slouží převodní rovnice A/D převodníku, kterou určuje výrobce čipu a vypadá:

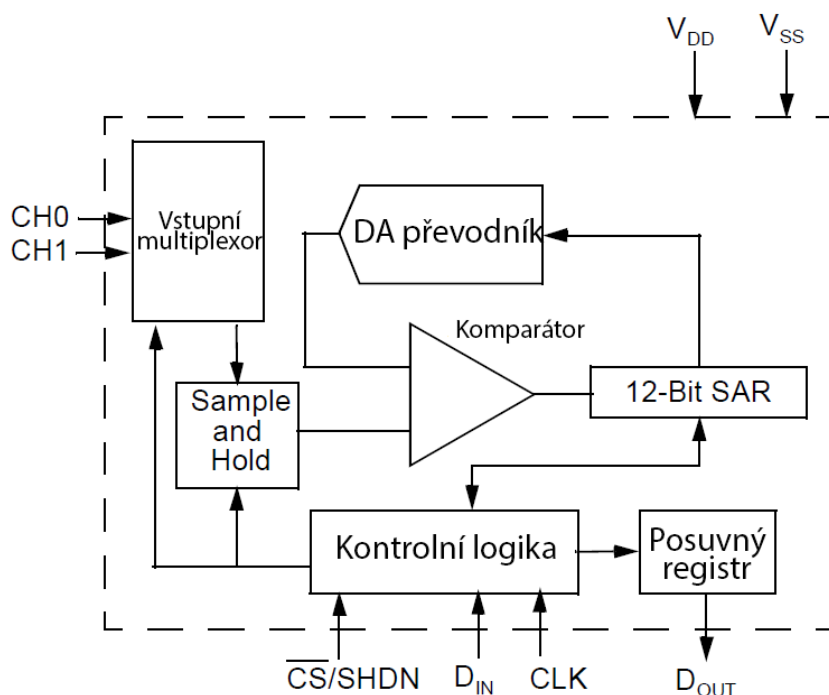
$$\text{Digitální výstupní hodnota} = \frac{4096 \cdot V_{\text{vstup}}}{V_{DD}} \quad (4)$$

kde *Digitální výstupní hodnota* je číslo, jenž převodník pošle ke zpracování mikropočítači, hodnota 4096 odpovídá maximální hodnotě, jenž může dvanáctibitové číslo zobrazit,  $V_{vstup}$  je analogová hodnota vstupního napětí a  $V_{DD}$  je hodnota napájecího napětí. [17]

Z rovnice 4 je patrné, že výstupní hodnota je vztažena k velikosti napájecího napětí. Napájecí napětí tak určuje měřicí rozsah A/D převodníku. Rovnici 4 lze upravit pro výpočet napětí ze získané hodnoty digitálního kódu. Pro tuto hodnotu po úpravě platí:

$$V_{vstup} = \frac{(Digitální\ výstupní\ hodnota) * V_{DD}}{4096} \quad (5)$$

Blokové schéma MCP3202 je zobrazeno na obrázku 10. A/D převodník obsahuje dva vstupní kanály, které jsou označeny CH0 a CH1. Napětí na těchto vstupech je posláno do multiplexeru, který posílá zvolená data dále. Blok Sample and hold zajišťuje udržení získaného vzorku signálu, aby při jeho změnách nedocházelo k chybné přeměně analogových dat na digitální. Komparátor porovnává získanou hodnotu vzorku s digitálně nastavenou hodnotu, kterou pomocí D/A převodníku přivádí na druhý vstup komparátoru. Toto zapojení společně se změnou úrovně referenčního signálu (vnitřně nastavená hodnota) provádí postupnou aproximaci získaného vzorku a převádí tak hodnotu jeho napětí na 12bitové číslo. Toto číslo je pomocí sériové komunikace a posuvného registru posláno do příslušné řídicí jednotky (pin D<sub>OUT</sub>), kde může být dále zpracováno. Správnost funkce a nastavení parametrů A/D převodníku zajišťuje kontrolní jednotka, která je ovládána pomocí vnější řídicí jednotky (CS/SHDN – pin pro určení zařízení, D<sub>IN</sub> – vstupní pin, CLK – pin pro přenos hodinového signálu). Piny V<sub>DD</sub> a V<sub>SS</sub> zajišťují přísun napájecího napětí pro A/D převodník a určují tak jeho napěťový rozsah a rychlost snímání vzorků.



Obrázek 10 Vnitřní blokové schéma A/D převodníku MCP3202 [17]

## 3.4 Raspberry Pi

Raspberry Pi je levný mikropočítač, který se dá zapojit na obvyklé periferie (monitor, klávesnice a myš), jako každý jiný stolní počítač. Kvůli své velikosti se hodí pro jak pro uživatele, kteří se učí programovat, tak i pro náročné projekty, vyžadující zkušenosti. Raspberry Pi je schopné provádět veškeré úkony jako stolní počítače od jednoduchého prohlížení internetu, až po zpracování různých procesů a hraní her. [18]

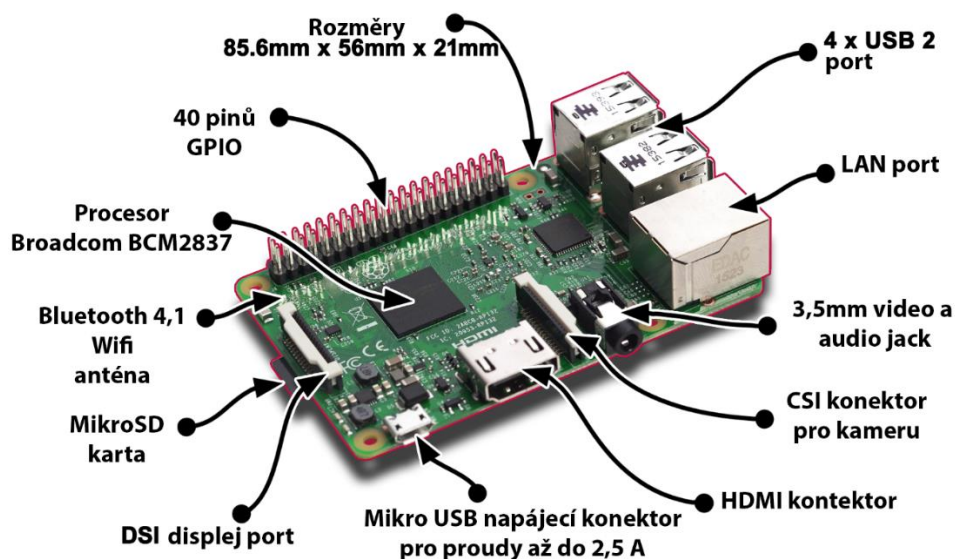
### 3.4.1 Raspberry Pi 3 model B

Raspberry Pi 3 model B je nejstarší model třetí generace mikropočítačů Raspberry Pi, který nahradil předchozí model Raspberry Pi 2 model B. I když se od přechodných modelů liší lepšími komponenty, jejich rozložení na desce zůstává stejné (obrázek 11). [18] Nově zajišťuje komunikaci pomocí standardu Bluetooth Low Energy (BLE) a poskytuje připojení na wifi síť. Napájení je zajištěno pomocí mikro USB (5V1/2,5 A). [20]

Jeho základem je čip Broadcom BCM2837, obsahující čtyři ARM Cortex-A53 jádra, pracující na frekvenci 1,2 GHz, dvoujádrový VideoCore IV grafický procesor a 1 GB LPDDR2 paměť. Operační systém zajištěný mikroSD kartou může být jak Linux (a jeho verze) tak i Windows 10 IoT. [19]

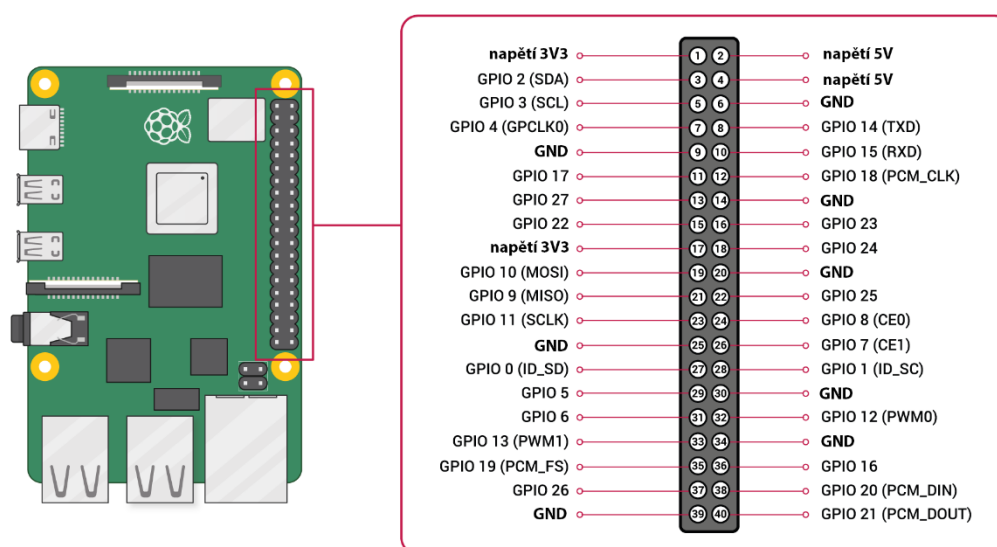
Grafický procesor poskytuje Open GL ES 2.0, Open VG (hardwarový urychlovač pro vektorovou grafiku), poskytuje vysoké rozlišení FullHD s rychlostí až 30 snímků za sekundu (1080p30). Díky standardu H.264 zajišťuje dobrou kompresi video signálu. Ta umožňuje zapojení do HDTV.

Pro ovládání mikropočítače a práci s externími periferiemi slouží čtyři USB porty (především komunikace pomocí klávesnice a myši) a konektor pro ethernet, zajišťující připojení na internet při nedostatečném wifi připojení. Součástí je také vestavěná anténa, která by měla zajišťovat přijímání signálu typu WLAN a Bluetooth i skrze zdi. [20]



Obrázek 11 Rozložení součástek a konektorů na desce Raspberry Pi 3 model B [20]

Pro práci s periferiemi, které nepoživají USB připojení, slouží 40 pinů osazených na desce (GPIO – general-purpose input-output). Jednotlivé piny jsou od sebe vzdáleny 2,54 mm. Při práci, 27 těchto pinů slouží pro komunikaci pomocí protokolů UART, I<sup>2</sup>C nebo SPI (označení GPIO). Dále dva piny slouží pro rozvod napětí o hodnotě 3,3 V a dva o hodnotě 5 V. Piny ID\_SD a ID\_SC jsou piny pro I<sup>2</sup>C ID EEPROM, které zajišťují správné nastavení všech 40 pinů (popřípadě linux driverů) po restartu desky. Díky své důležitosti by se neměly používat. Zbylé piny poté slouží jako zem GND. Rozložení těchto pinů je vidět na obrázku 12. [20]



Obrázek 12 Zobrazení funkčnosti pinů mikropočítače Raspberry Pi 3b [21]

I když není Raspberry Pi 3 model B nejnovější model od výrobce Raspberry Pi a ani nejnovější model ze série 3 B, měl by poskytovat dostatečný výkon a funkce pro sestavení všech potřebných programů a zajišťovat přehlednou práci při řešení úloh.

### 3.4.2 Nastavení mikropočítače raspberry Pi 3b

Pro správnou funkci a zajištění komunikace mezi mikropočítačem Raspberry Pi 3b a příslušnými senzory je nutné provést jisté operace. Jedná se o nainstalování všech potřebných knihoven, které jsou využívány v programech pro komunikaci a zpracování dat. Dále je pak nutné systémově povolit komunikaci příslušných sériových protokolů, a to I<sup>2</sup>C a SPI.

#### Instalace knihoven

Programy, které jsou využity jsou napsány v programovém jazyce Python. Instalace potřebných knihoven může probíhat dvěma způsoby.

První možnost nastává, pokud se potřebná knihovna nachází v archívech, které náleží nainstalovanému operačnímu systému (Raspbian). Pokud tato možnost dostupná, nainstalování knihovny je možno provést pomocí dvou příkazů. Do terminálu operačního

systému je nutné zapsat příkazy:

```
sudo apt update  
sudo apt install python3-<název balíčku>
```

*sudo* je zkratka z anglického sousloví „super user do“. Tento příkaz zajišťuje provedení následující operace jako hlavní uživatel (uživatel, který má přístup k celému operačnímu systému a všem souborům).

Příkaz *apt* je z anglického sousloví „Advance packaging tools“. Tento příkaz slouží jako nástroj pro přístup k různým balíčkům z různých ověřených zdrojů a slouží především k instalaci, aktualizaci nebo odstranění těchto balíčků. [22]. Za tímto příkazem již následuje, co se má provést. Nejdříve je pomocí příkazu *update* aktualizovaný celý operační systém (Raspbian). Druhý příkaz *install* slouží pro stažení a instalaci balíčku. Následuje určení, o jaký balíček se jedná. *Python3-* určuje balíček pro programovací jazyk Python verze 3 (pro verzi 2 se balíček značí pouze *python-*). Název balíčku poté určuje, o jakou knihovnu se jedná.

Pokud se potřebný balíček nenachází v archívech operačního systému, je pro jeho instalaci nutné využít instalační nástroj *pip*. Nástroj *pip* využívá přístup ke stránce Python Package Index [23]. Tato stránka slouží pro hledání, instalaci a publikaci balíčků pro programovací jazyk python. Nástroj *pip* je možné nainstalovat pomocí příkazu:

```
sudo apt install python3-pip
```

kde *python3-* značí, pro jakou verzi jazyka python bude nástroj *pip* nainstalován. Následná instalace potřebných balíčků je provedena pomocí příkazu:

```
sudo pip3 install <název balíčku>
```

kde *pip3* značí, že se jedná o jazyk Python verze 3. Pro verzi 2 by byl zápis pouze *pip*. [24]

Při instalaci (především pomocí instalačního nástroje *pip*) je nutné připojení k internetu, a to buď pomocí ethernetového kabelu nebo wifi sítě. Připojení k internetu je také nutné pro příkaz *update*, pro správné aktualizování operačního systému.

### **Povolení protokolů I<sup>2</sup>C a SPI pro sériovou komunikaci**

Konfigurace a povolení těchto protokolů je možné dvěma způsoby.

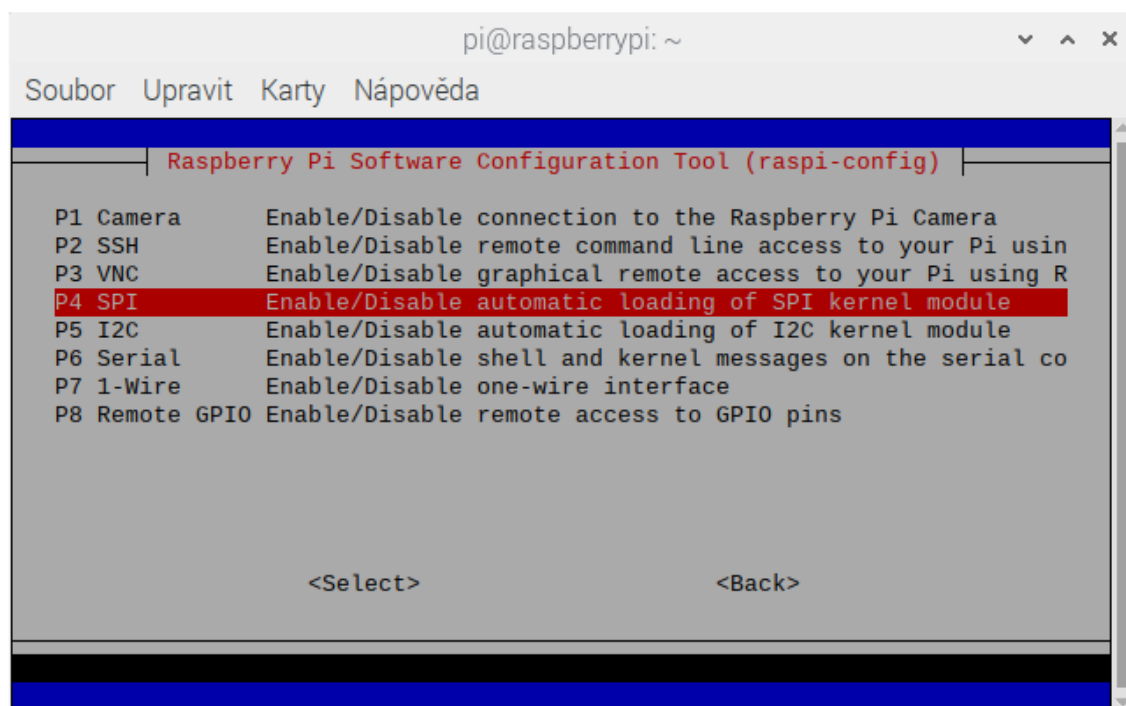
Pokud je mikropočítač Raspberry Pi v grafickém módu (přístup na plochu) postup je následující. V levém horním rohu je nutné rozbalit nabídku raspberry pi (aplikační menu), zde rozbalit záložku Volby (pomocí kurzoru myši) a zvolit možnost Raspberry Pi Configuration (nastavení Raspberry Pi). Po zvolení se rozklikne okno, ve kterém se zvolí záložka Interfaces (rozhraní). Zde se u protokolu SPI a I<sup>2</sup>C zvolí možnost Enable (povolit). Nastavení se potvrdí tlačítkem OK. Po povolení těchto protokolů je pro správnou činnost doporučený restart mikropočítače Raspberry PI. Tento postup se může lišit pro jiné operační systémy než Raspbian.

Při použití terminálu (negrafické rozhraní) je nutné napsat příkaz

```
sudo raspi-config
```

který otevře okno pro systémové nastavení mikropočítače. Zde se pomocí šipek na klávesnici zvolí možnost Interfacing Options (možnosti rozhraní, možnost číslo 5) a následně se povolí protokoly SPI a I<sup>2</sup>C (obrázek 13). Pomocí pravé šipky se zvolí příkaz DOKONCI. Když se poté systém zeptá na možnost restartu, pro správné nastavení zvolte ANO. [25]





Obrázek 13 Povolení sériové komunikace SPI a I2C pomocí terminálu na mikropočítači Raspberry Pi



## 4 NÁKLON

### 4.1 Teorie výpočtu náklonu a rotace

Pomocí měření změny statického zrychlení lze zjistit hodnotu vzniklého náklonu a rotace. Tento efekt je způsoben gravitačním polem Země, které změní polohu pohyblivých elektrod v kapacitním senzoru. Tato změna se pak projeví jako změna zrychlení daných os. Pokud poloha akcelerometru zůstává stejná, působení gravitace vyvolává stále stejnou gravitační sílu, elektrody zůstávají na stejném místě a měřené zrychlení se nemění (statické zrychlení).

Pomocí goniometrických rovnic pak dostaneme rovnice pro náklon a rotaci (4 a 5), které se dají vypočítat pomocí statického zrychlení v gravitačním poli. Určení, zda se jedná o náklon nebo rotaci záleží na úhlu, ze kterého se na akcelerometr díváme.

Rovnice náklonu a rotace jsou pak následující:

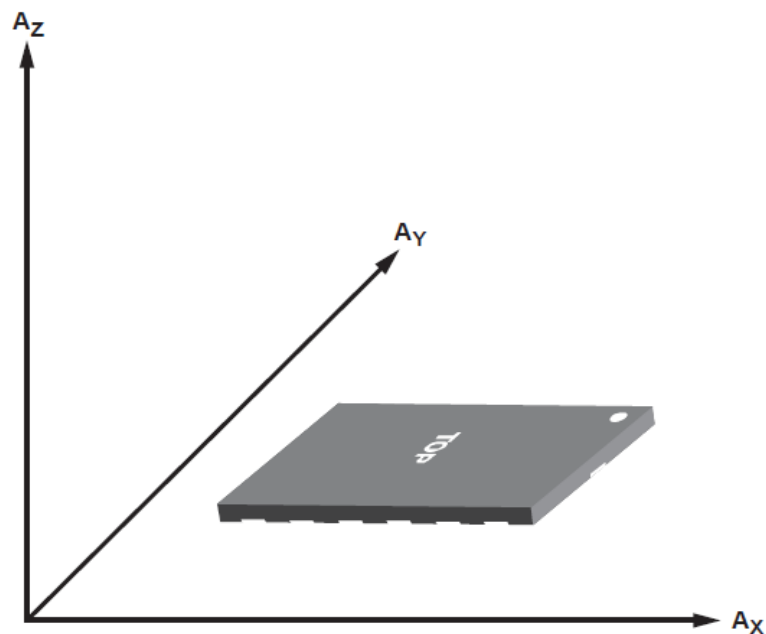
$$\tan \theta_{xyz} = \frac{-G_x}{\sqrt{G_y^2 + G_z^2}} * \frac{180}{\pi}, \quad (6)$$

$$\tan \theta_{xyz} = \frac{G_y}{\sqrt{G_x^2 + G_z^2}} * \frac{180}{\pi}, \quad (7)$$

kde  $\theta_{xyz}$  je úhel náklonu nebo rotace a  $G_x, G_y, G_z$  jsou změřené hodnoty zrychlení v jednotkách gravitační konstanty  $g$  ( $g = 9,806 \text{ m*s}^2$ ).

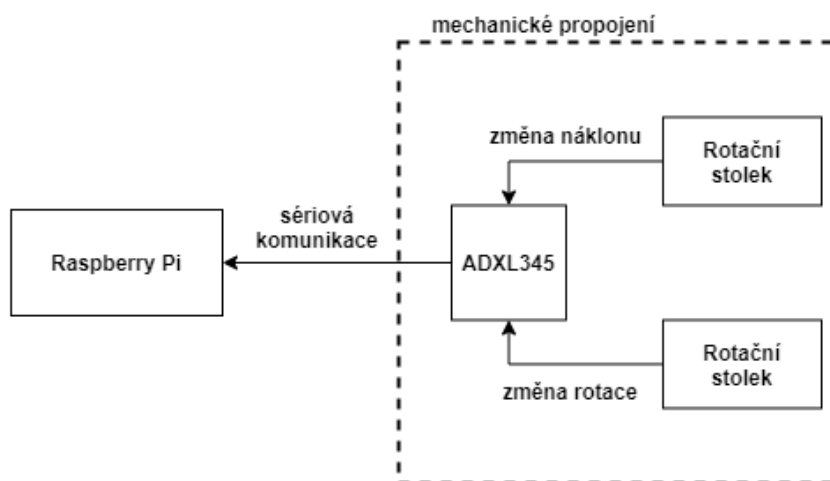
Tyto rovnice platí při splnění dvou podmínek:

1. Na akcelerometr nepůsobí žádná jiná síla, která by způsobovala vznik dynamického zrychlení
2. Počáteční orientace akcelerometru je vodorovná a gravitační pole působí rovnoběžně s osou z, obrázek 14. [26]



Obrázek 14 Určení směru jednotlivých os u tříosého akcelerometru [15]

Blokové schéma pro měření náklonu a rotace je zobrazeno na obrázku 15. Tříosý akcelerometr ADXL345 změří hodnoty statického zrychlení všech os. Změny těchto hodnot budou zajišťovat dva rotační stolky, kde jeden bude určen pro změnu náklonu a druhý rotace. Po potřebné změně se pomocí sériové linky (protokol I<sup>2</sup>C) a mikropočítače Raspberry Pi zjistí digitální hodnoty vzniklých zrychlení. Tyto hodnoty se pomocí vytvořeného programu zpracují (program zajišťuje také komunikaci s akcelerometrem).



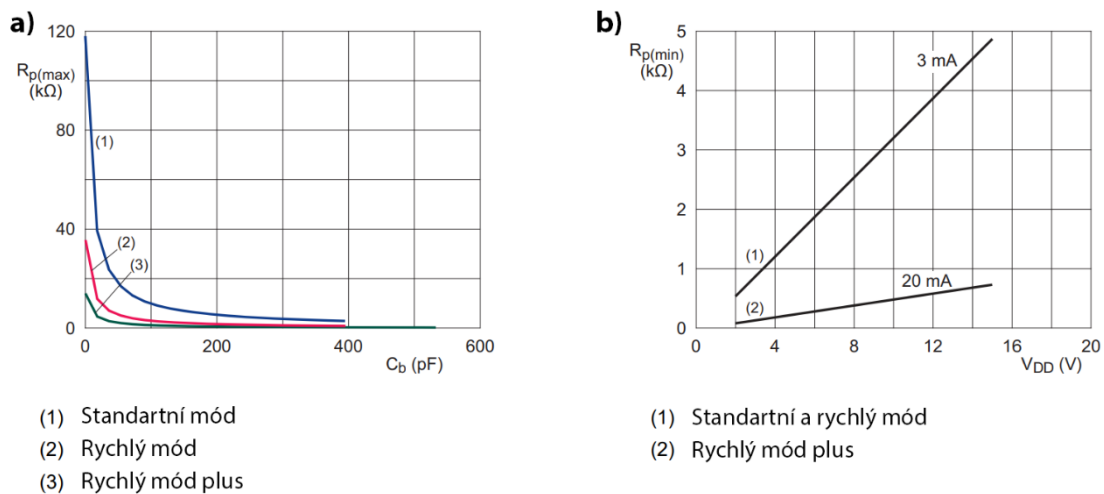
Obrázek 15 Blokové schéma principu úlohy pro měření náklonu a rotace

## 4.2 Zapojení

### 4.2.1 Deska

Základ tvoří čip značen ADXL345 (popis v kapitole 3.1.), který je umístěn v pouzdře typu LGA. Díky možnosti napájecího napětí 3,3 V z pinu mikropočítače Raspberry Pi, lze čip napájen přímo tímto napětím a není nutné dalších regulátorů. Mezi vstupním napětím a zemí jsou umístěny dva kondenzátory, a to tantalový  $C_1$  o hodnotě 2,2  $\mu\text{F}$  a keramický  $C_2$  o hodnotě 100 nF. Tyto kondenzátory plní funkci filtrace a filtrují tak rušivé signály, které mohou vznikat na napájecím napětím.

Na výstupních pinech určených pro sériovou komunikaci, piny SDA/SDI/SDIO, SCL/SCLK a  $\overline{\text{CS}}$ , jsou umístěny pull-up rezistory s hodnotami 4,7 k $\Omega$  a 10 k $\Omega$ . Tyto hodnoty byly zvoleny z grafů, na které poukazuje výrobce čipu (obrázek 16). Takto zapojené rezistory zajišťují hodnotu logické 1 (3,3 V) na těchto pinech v době jejich nečinnosti (stav vysoké impedance). Piny SDA/SDI/SDIO a SCL/SCLK jsou využité pro komunikaci pomocí protokolu I<sup>2</sup>C a hodnota logické 1 na pinu  $\overline{\text{CS}}$  povoluje komunikaci pro daný čip.



Obrázek 16 Závislost zvolené hodnoty pull-up rezistoru na a) kapacitě sběrnice b) napájecího napětí sběrnice

Obrázek 16a značí závislost maximální hodnoty pull-up rezistoru na kapacitě sběrnice. Závislost odpovídá rovnici:

$$R_{p(max)} = \frac{t_r}{0,8473 * C_b}, \quad (8)$$

kde  $R_{p(max)}$  představuje maximální hodnotu pull-up rezistoru,  $t_r$  je maximální doba náběhu pro daný signál (SDA, SCL) a  $C_b$  je kapacita sběrnice.

Pro minimální hodnotu pull-up rezistoru souží závislost této hodnoty na použitém napětí sběrnice. Tato závislost je vidět na obrázku 16b. Pro vykreslení této závislosti platí vztah:

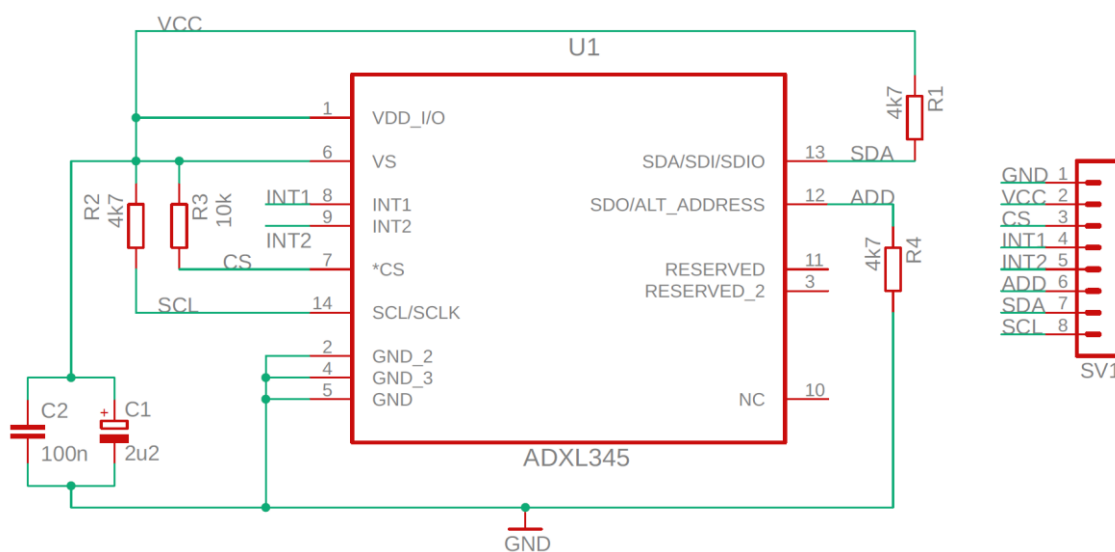
$$R_{p(min)} = \frac{V_{DD} - V_{OL(max)}}{I_{OL}}, \quad (9)$$

kde  $R_{p(min)}$  je minimální hodnota pull-up rezistoru,  $V_{DD}$  je napětí sběrnice,  $V_{OL(max)}$  je maximální hodnota napětí při stavu logické 0 a  $I_{OL}$  představuje minimální proud při stavu logické 0. [27]

Pin SDO/ALT ADDRESS je oproti ostatním logickým pinům přiveden přes rezistor na zem (pull-down rezistor). Toto připojení zajišťuje hodnotu logické 0, na tomto vstupu, ve stavu vysoké impedance (pokud není změněna signálem z mikropočítače) a určuje tak hodnotu adresy celého akcelerometru. V tomto zapojení (logická 0) tak adresa odpovídá hodnotě 0x53. Tato hodnota je určená výrobcem a je uvedena v katalogovém listě.[15] V případě přivedení pinu SDO/ALT ADDRESS na hodnotu logické 1, odpovídá hodnota adresy číslu 0x1D. Všechny vstupy a výstupy, včetně pinů pro úlohu nepotřebných, mohou být vyvedeny z desky pomocí drátků, jenž se dají připojit na potřebné GPIO piny mikropočítače Raspberry Pi.

Hodnota kondenzátorů  $C_1$  a  $C_2$ , jejichž funkce je dodatečné filtrování napájení, které poskytuje mikropočítač Raspberry Pi, byly zvoleny jako hodnoty, které doporučuje výrobce akcelerometru ADXL345.

Schéma a návrh desky bylo provedeno v programu EAGLE. Schéma zapojení je zobrazeno na obrázku 17. Layout vytvořených desek je zobrazen v příloze A.



Obrázek 17 Schéma zapojení desky s akcelerometrem ADXL345

#### 4.2.2 Propojení akcelerometru ADXL345 a mikropočítače Raspberry Pi

Z důvodu využití protokolu I<sup>2</sup>C jsou pro komunikaci potřeba pouze dva kabely. Jeden kabel slouží pro propojení výstupu SDA (komunikační sběrnice pro přenášené data) akcelerometru (zelený kabel). Druhý kabel slouží pro propojení výstupu s označením SCL (hnědý kabel), který slouží k přenosu hodinového signálu a zajišťuje samotnou

komunikaci (synchronní sériová komunikace). Tyto kabely jsou připojeny na piny GPIO mikropočítače Raspberry Pi pod identickým označením SDA a SCL (obrázek 12).

Pro připojení napájení akcelerometru slouží vstupy značeny VCC a GND. Na vstup VCC (červený kabel) je nutné připojit napětí o velikosti 3,3V. Z těchto důvodů je vstup VCC připojen na kterýkoli pin GPIO mikropočítače Raspberry Pi, který toto napětí dodává (obrázek 12). Vstup GND (černý kabel) se propojí s jakýmkoli pinem GND, který Raspberry Pi poskytuje.

Takto propojené napájecí a komunikační kabely zajišťují ovládání a práci s akcelerometrem ADXL345 pomocí mikropočítače Raspberry Pi.

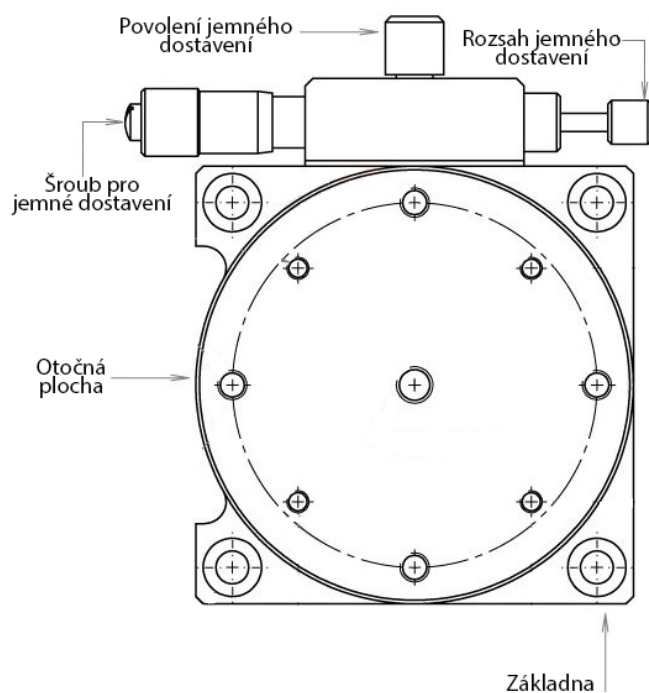
### 4.2.3 Systém změny náklonu a rotace

Změnu náklonu a rotace zajišťují dva manuální rotační stolky s označením R60A-L, který je zobrazen na obrázku 18. Jedná se o stolek velikosti 60 mm na 60 mm s otočným kruhovým středem. Otočný kruhový střed je označen hodnotami úhlů, a to v rozsahu 0° až 360° po jednom stupni. Stolek je opatřený šroubem pro jemné dostavování, který má, kterým lze nastavovat hodnotu po 0,2°.

Návrh částí aktuátoru pro změnu náklonu a rotace jsou zobrazeny v příloze C. Rotační stolky jsou pomocí podpůrného držáku a děr, které obsahuje rotační kruhový střed, připevněny kolmo na sebe. Následně je základní stolek připevněn na podpůrnou základnu pro udržení správné hodnoty a zlepšení manipulace a nastavení. Na výstupní stolek (volné díry na šrouby) je připevněn držák pro desku s akcelerometrem ADXL345. Každý stolek v takto spojené soustavě poté zajišťuje změnu některého z měřených parametrů, tedy jeden stolek zajišťuje změnu náklonu, zatímco druhý zajišťuje změnu rotace desky s akcelerometrem.

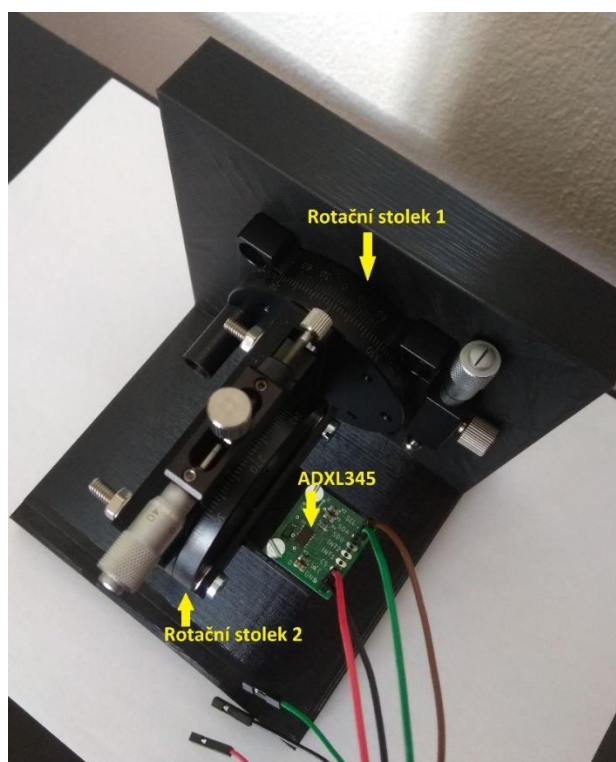
Pro vyčtení správné hodnoty slouží dvě stupnice umístěné ve výřezu v rotačním stolku. Jedna stupnice se nachází na otočné části a určuje tak změnu úhlu o požadované velikosti ve stupních (jeden dílek stupnice odpovídá 2°). Druhá stupnice je umístěna na základně stolku a slouží k přesnému určení úhlu. Cejchování stupnice na základně stolku je provedeno tak, že hodnoty 30 a 60 odpovídají hodnotě 1° a 2° a jeden dílek této stupnice odpovídá hodnotě deseti minut). Výsledná přesná hodnota úhlu se odečte jako hodnota, kde je požadovaný úhel (otočná stupnice) nachází v rovině s příslušnou minutovou indikací. Stolek také obsahuje přesné dostavování úhlů. Tato možnost se zprovozní po utáhnutí šroubu Povolení jemného dostavení (obrázek 18). Při dotažení není možné s otočnou částí pohybovat ručně, ale pouze pomocí šroubu pro jemné dostavení (obrázek 18). Jemné dostavení je cejchováno na tomto šroubu a při jednom celém otočení šroubu se hodnota úhlu změní o jeden stupeň.

Pro jemné dostavení je vhodné nastavení stolku co nejpřesněji k potřebnému úhlu, následně šroub pro jemné dostavení nastavit na hodnotu 0 a utáhnout šroub na povolení jemného dostavení. Následně se dá jemně měnit náklon nebo rotace pomocí příslušného šroubu a pomocí stupnicích ve výřezu lze odečíst přesnou hodnotu.



Obrázek 18 Schéma rotačního stolku RL60-A

Na obrázku 19 je zobrazen vytvořený aktuátor pro změnu náklonu a rotace společně s upevněnou deskou, která obsahuje akcelerometr ADXL345.



Obrázek 19 Vytvořený aktuátor pro změnu náklonu a rotace

## 4.3 Popis použitých knihoven

### 4.3.1 SMBUS

Komunikaci mezi akcelerometrem ADXL345 a mikropočítačem Raspberry Pi zajišťuje sériový komunikační protokol I<sup>2</sup>C. Pro tento protokol a programovací jazyk Python je požita knihovna funkcí, které zajišťují správnou komunikaci. Tato knihovna se jmenuje `smbus`. Z této knihovny jsou použité tři funkce. První funkce slouží k inicializaci samotného I<sup>2</sup>C v Raspberry Pi.

```
smbus.SMBus(1)
```

Jedná se o funkci, která má jako parametr funkce číslo, kterou sběrnici má používat (Raspberry Pi obsahuje pouze jednu sběrnici, která má číslo 1). Funkce se používá hodnota proměnné v programu, která zajišťuje zlepšení práce a přehlednosti programu.

Další použitá funkce slouží pro zápis slova o velikosti 1 byte (8 bitů) do registru v senzoru ADXL345 a její zápis je:

```
write_byte_data(Addr, Comm, Data)
```

Funkce pracuje se třemi vstupními parametry, které jsou:

*Addr* – zpravidla 7bitová adresa (může být i 10bitová) zařízení, které je připojeno na piny pro sériovou komunikaci I<sup>2</sup>C. Pro senzor ADXL345 je hodnota této adresy rovna 0x53 při uzemnění pinu SDO/ALT ADDRESS (logická 0) a hodnota 0x1D pro nastavení tohoto pinu na úroveň logické 1.

*Comm* – 8bitové slovo, které slouží pro určení příkazu, který se má provést. Ve většině případů toto slovo určuje registr, do kterého se požadovaná data mají zapsat.

*Data* – jedná se o 8bitové slovo nesoucí určitou informaci, které může sloužit k nastavení zařízení nebo k průběhu další komunikace.

Po předání správných parametrů a jejího zavolání, funkce vykoná zápis. Ten probíhá: Posláním start bitu (1 bit) na pin pro komunikaci SDA, poslání 7bitové adresy (*Addr*) doplněné o 1 bit, které určuje, zda se jedná o zápis nebo čtení (logická 0 pro zápis). Po této inicializaci následuje potvrzovací bit (1 bit, logická 0), který slouží pro správný chod komunikace. Poté následuje určení, do kterého registru se bude zapisovat (8bitové slovo *Comm*) a znovu potvrzovací bit. Jako poslední se pošlou data, která jsou zapsány do zvoleného registru. Po přenesení dat se pošle poslední potvrzovací bit, za kterým následuje stop bit (1 bit), který ukončí komunikaci. [28]

Třetí použitá funkce z knihovny `smbus` slouží pro čtení 8bitového slova ze senzoru ADXL345:

```
read_byte_data(Addr, Comm)
```

Funkce obsahuje dvě vstupní hodnoty *Addr*, a *Comm*. Tyto hodnoty plní stejnou funkci jako při zápisu, tedy určují adresu zařízení, se kterým se bude komunikovat (*Addr*) a určují registr, ze kterého má funkce vyčíst 8bitovou hodnotu (*Comm*).

Čtení je rozděleno do dvou kroků. Nejdříve je nutné určit, ze kterého registru se bude číst. Začátek tedy určuje start bit, následuje adresa s bitem určující zápis (logická 0) a potvrzovací bit. Poté se pošle hodnota registru (*Comm*) a následuje potvrzovací bit. Po tomto bitu nastává nová komunikace začínající start bitem a adresou. Po poslání adresy je poslán byt určující čtení ze zařízení (logická 1) a potvrzovací bit. Po tomto bytu nastává

vyčtení dat ze zařízení, které se uloží do programově zvoleného registru (proměnné) a následuje opačný potvrzovací bit (logická 1) který určuje, že čtení proběhlo v pořádku. Po tomto bitu následuje jednobitový stop bit. [28]

### 4.3.2 Tkinter

Pro vizuální zpracování získaných hodnot z akcelerometru byla použita knihovna tkinter. Tato knihovna pracuje s widgety a dokáže tak hodnoty zpracovat do přehledného okna. Použité funkce z této knihovny jsou:

```
Tk(screenName=None, baseName=None, className='Tk', useTk=1)
```

Funkce, která vytváří hlavní okno pro přidání dalších widgetů. Tato funkce může pracovat také bez předání jakéhokoli parametru. V tomto případě se vytvoří bezejmenné okno. Návrátová hodnota funkce je vytvořené hlavní okno. Pro lepší orientaci v programu a předávání tohoto okna jako parametr se tato návratová hodnota ukládá do zvolené proměnné.

```
Label(master, option=value)
```

Jedná se o funkci, která ve zvoleném okně umožní zobrazit textové pole. Vstupní parametr *master* představuje proměnou určující okno (proměnná vytvořená pomocí funkce Tk()). Parametr *option* představuje různé parametry, které nastavují obsah a vzhled textového pole. Některé z mnoha nastavujících parametrů mohou být:

Text – parametr zajišťující zobrazení potřebného textu

Textvariable – Proměnná, jenž v sobě obsahuje text. Tato proměnná musí být formátu, jenž poskytuje knihovna Tkinter, tedy StringVar(). Zajišťuje zobrazení textu, vhodné pro měnící se text.

Width – určuje délku příslušného textového pole v pixelech

Height – určuje výšku příslušného textového pole v pixelech

Padx/pady – Vzdálenost textu od hrany vytvořeného textového pole v pixelech

Borderwidth – Určuje šířku ohraničení textového pole (většinou 1 nebo 2 pixely)

Relief – určuje 2D nebo 3D efekt kolem ohraničení textového pole

```
Radiobutton(master, option=value)
```

Funkce vytvoří tlačítko v podobě zaškrťovacího kolečka. Jedná se o tlačítko vhodné pro výběr z několika možností. Vstupní parametr *master* představuje okno, se kterým bude tlačítko spojeno (okno vytvořené například funkcí Tk()). Parametr *option* představuje proměnné, které modifikují jeho vzhled a určují, jaký příkaz se při vybrání tlačítka má provést. [29]

```
<x>.grid(row=value, column=value)
```

Jedná se o funkci, které modifikuje již vytvořené widgety, jako jsou například tlačítka a textová pole (parametr <x> před samotnou funkcí). Tato funkce uděluje těmto widgetům místo ve vytvořeném okně. Vstupní parametry *row* a *column* určují místo pomocí tabulkového systému, kde hodnota *row* určuje řádek, na kterém se bude widget nacházet. Hodnota *column* poté určuje sloupec umístění příslušného widgetu <x>. Při určování řádků a sloupců je nutné vědět, že pokud se ve sloupci nebo řádku nenachází žádný widget (textové pole, tlačítko a jiné), grafické rozhraní tento sloupec (řádek)



ignoruje a dva widgety, které by měly mít mezi sebou místo jsou umístěny těsně na sebe. Dalšími parametry funkce mohou být proměnné, které určují například roztáhnutí widgetu přes více řádků/sloupců nebo odskok textu, popřípadě obrázku od okraje widgetu.

```
<master>.update()
```

Funkce obnovuje okno, na které ukazuje proměnná `master`. Toho se využívá například při změně určitých parametru vlivem dalších funkcí programu, popřípadě zmáčknutím tlačítka, které vykoná určitou funkci.

```
protocol("WM_DELETE_WINDOW", callback)
```

Funkce zajišťuje zavolání funkce (callback) při nastání určitého protokolu (například "WM\_DELETE\_WINDOW"), zajišťující komunikaci mezi uživatelem a vytvořeným oknem. Protokol "WM\_DELETE\_WINDOW" nastává při zavření okna. Další protokoly mohou být například "WM\_TAKE\_FOCUS" (požití při zobrazení více oken a přepínání do jakého okna má být poslán vstup od uživatele, například z klávesnice) nebo "WM\_SAVE\_YOURSELF", které slouží pro upozornění uživatele od programového manažera oken, aby si uložil aktuální stav. Toto upozornění může nastat například při nečekaném ukončení programu. [30]

### 4.3.3 Math

Jedná se standardizovanou knihovnu programovacího jazyka python. Tato knihovna umožňuje přístup k matematickým funkcím definovaného pomocí standartu C. Knihovna slouží pro práci s reálnými čísly, nikoli s komplexními čísly. Návrátová hodnota funkcí, jenž tato knihovna používá je typu float (pokud není určeno jinak). Knihovna byla využita z důvodu výpočtu náklonu a rotace a zajišťuje tak využití funkce pro výpočet arkus tangens, druhé odmocniny, druhé mocniny a použití konstanty  $\pi$ .

### 4.3.4 Time

Jedná se o modu, který poskytuje přímo programovací jazyk python. Tento modul slouží pro práci s časem a časově založenými úkoly. Tento modul je využit především pro funkci `time.sleep(x)`. Tato funkce slouží k pozastavení programu na určitou dobu, kde `x` je v jednotkách vteřin. Funkce je použita, aby se příkazy zaslané z mikropočítače Raspberry Pi správně zapsali do akcelerometru ADXL345.

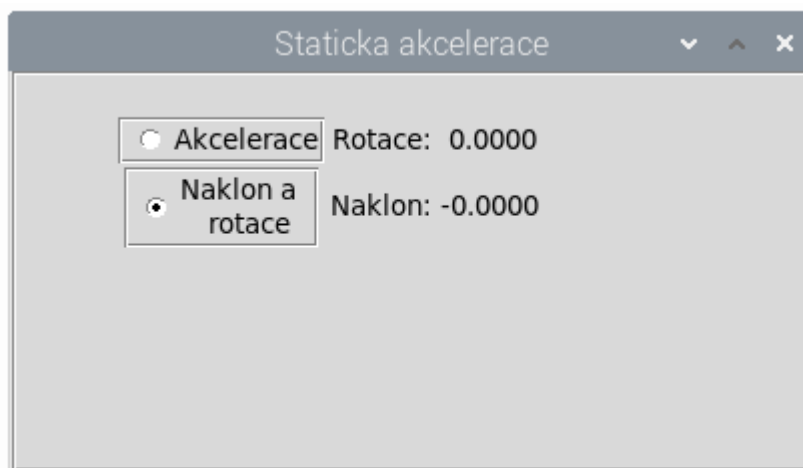
## 4.4 Vytvořené funkce pro sepsání programu

Následující funkce se nacházejí v knihovně `kADXL345.py`. Knihovna je napsaná a určena pro programovací jazyk python. Pro správnou funkčnost je nutné umístit knihovnu do stejné složky jako soubor programu, jenž budou žáci programovat sami pomocí těchto funkcí. Díky tomu mohou žáci importovat tuto knihovnu pomocí příkazu `import`, jako normální knihovny, které poskytuje programovací jazyk python.

```
Pocatecni_inicializace()
```

Funkce bez vstupních parametrů. Zajišťuje správné vytvoření okna pro umístění widgetů a potřebných měřených hodnot. Vytváří ono pomocí funkce `Tk()`, následně vytváří proměnné s formátem `IntVar()` a `StringVar()` (proměnné určené pro poslání

správného druhu do tkinter funkce). Následně vytváří tři textová pole pro zobrazení výsledků (funkce Label) a dvě kulatá tlačítka pro možnosti volby mezi zobrazenými hodnotami (funkce Radiobutton). Těmto widgetům také přiřazuje místo pomocí funkce grid. Jako poslední zajišťuje platnost protokolu "WM\_DELETE\_WINDOW" (funkce protocol), který při zavření okna zavolá funkci s názvem Ukonceni. Funkce pocatecni\_inicializace má návratovou hodnotu proměnou root, která odkazuje na vytvořené okno pomocí funkce Tk, se všemi vytvořenými widgety. Všechny vytvořené a nastavené prvky v této funkci poté vytváří zobrazovací okno pro měření, které je vidět na obrázku 20.



Obrázek 20 Vytvořené okno pro zobrazení naměřených hodnot

`Ukonceni()`

Funkce Ukonceni je volána při zavření zobrazovacího okna. Tato funkce mění hodnotu globální proměnné kontrola\_ukonceni na hodnotu 1. Tato hodnota je v jiných funkcích zpracována a zajišťuje ukončení programu a správné zničení vytvořeného okna.

`Zobraz(root, x, y, z)`

Vstupní parametry této funkce jsou proměnná odkazující na vytvořené okno (parametr *root*) a hodnoty změřené akcelpace pro všechny tři osy akcelerometru ADXL345 (parametry *x*, *y*, *z*, musí být poslány v tomto pořadí). Funkce nejdříve zkontroluje, zda nebyla změněná hodnota globální proměnné kontrola\_ukonceni na hodnotu 1 (zavření okna), pokud ano funkce nic neprovede a vrací návratovou hodnotu 1. Pokud tato podmínka nebude splněna, funkce zjistí hodnotu globální proměnné *r* (parametr určující, které tlačítko je potvrzeno). Podle hodnoty této proměnné se do dříve vytvořených textových polí vloží hodnoty změřené akcelpace nebo vypočítaného náklonu a rotace podle vzorce 6 a 7 ze vstupních parametrů. Po nastavení těchto proměnných je zavolána funkce update, která změní hodnoty ve zobrazovacím okně. Návratová hodnota této funkce je při normálním průchodu funkcí 0.

`Konec_mereni(indikace, root)`

Funkce sloužící ke správném ukončení programu a zničení okna. Zjišťuje, zda vstupní parametr *indikace* je roven hodnotě 1. Pokud ano funkce zničí okno, na které odkazuje parametr *root* a následně ukončí celý program.

`Precti(bus, adresa0, adresa1)`

Funkce slouží pro přečtení zrychlení měřeného akcelerometrem ADXL345. Funkce pracuje s protokolem sériové komunikace I<sup>2</sup>C a knihovnou `smbus`. Vstupním parametrem *bus* se určí, která sběrnice se pro komunikaci bude používat (proměnná vytvořená funkcí `smbus.SMBus`). *Adresa0* a *adresa1* slouží pro učení, které dva registry se z akcelerometru ADXL345 mají přečíst (důvodem dvou registrů je, že výstupní hodnota měřené akcelerace je desetibitové číslo, které je uloženo na dvou osmibitových registrech, se kterými akcelerometr pracuje. Pomocí vstupních parametrů a funkce `read_byte_data(0x53, adresa0)`, kde konstanta 0x53 představuje adresu senzoru, jsou tyto dva registry sečteny a jejich hodnoty jsou zpracovány tak, aby návratová hodnota odpovídala zrychlení v jednotkách g.

```
Inicializace_adxl345(bus, x_offset, y_offset, z_offset)
```

Funkce slouží pro uvedení a nastavení akcelerometru do měřicího stavu a nastavení offsetových registrů pro měření správné hodnoty zrychlení. Vstupní parametr *bus* určuje sběrnici komunikace I<sup>2</sup>C, která byla nastavena pro komunikaci se senzorem pomocí knihovny `smbus` a funkce `smbus.SMBus`. Další tři parametry zajišťují adresy offsetových registrů pro jejich změnění. Nejdříve jsou ověřeny správně zadané adresy offsetových registrů (musí být postupně pro x, y, z). Následuje inicializace akcelerometru zapsáním hodnoty 0x08 do registru 0x2D (nastavení měřicího módu) a vynulování offsetových registrů pomocí funkce `write_byte_data`. Poté je přečtena hodnota aktuálního zrychlení (počáteční poloha senzoru) z jedné osy. Tato hodnota je přepočítána podle rovnice (x), který uvádí výrobce. Následuje kontrola, zda je offset hodnota záporná (pokud ano převedena na správnou hodnotu pro zapsání do registru) a je zapsána do příslušného offset registru dané osy. Tento krok se opakuje pro zbylé dvě osy. Mezi jednotlivými zápisy hodnot do registrů akcelerometru nastává programové čekání. Důvodem této prodlevy je dostatečný čas pro zapsání správné hodnoty na správné místo a zajištění správného nastavení.

```
offset = -(round(data/4))
```

*Offset* představuje výslednou hodnotu ve formě zaokrouhleného celého čísla, které se zapíše do příslušného registru pro osu, na které byly naměřeny hodnoty zrychlení. *Data* představují změřenou akceleraci v počátečním stavu akcelerometru. Funkce `round()` slouží k zaokrouhlení výsledku po dělení, aby se do registru mohlo zapsat celé číslo. Hodnota dělení 4 je hodnota, jenž udává výrobce akcelerometru ADXL345 a představuje hodnotu, která je způsobena rozdílností citlivostí offset registru (15,6 mg/LSB) a registru pro snímání aktuální hodnoty (3,9 mg/LSB). [15].

#### 4.4.1 Postup pro studenty k sestrojení programu

1. Vytvořte si soubor s příponou `.py` (například textový editor, Thonny Python IDE) ve stejné složce, kde se nachází soubor s vytvořenými funkcemi `kADXL345.py`.
2. Nejdříve je nutné importovat potřebné knihovny, se kterými se bude pracovat. Tyto knihovny mají název `smbus` a `kADXL345`. Tyto knihovny jsou do souboru importovány pomocí příkazu:

```
import <název knihovny>
```

3. Poté je nutné vytvořit proměnné, do kterých jsou uloženy odkazy na vytvořené zobrazovací okno (funkce `pocatecni_inicializace()` z knihovny `kADXL345`) a odkaz na používanou sběrnici sériové komunikace (funkce `SMBus(1)` z knihovny `smbus`). Přístup k těmto funkcím probíhá přes odkaz na knihovnu ve tvaru:

```
proměnná = <název knihovny>.<funkce>
```

4. Před napsáním funkce `while`, která bude měřit a zobrazovat aktuální hodnotu je nutné provést nastavení akcelerometru ADXL345. Nastavení se provede zavoláním funkce *Inicializace\_adxl345* (kapitola 4.2). Vstupními parametry bus je dříve vytvořená proměnná, odkazující na používanou sběrnici pro komunikaci. Zbylé vstupní parametry jsou hodnoty adres offset registrů, obrázek 7.
5. Následuje vytvoření funkce `while` s parametrem `True`. Parametr `True` vytváří nekonečnou smyčku a zajišťuje tak trvalý chod funkce. Po zapsání funkce `while` je nutné psát další příkazy vzdáleny od začátku řádku o čtyři mezery (nebo jedno zmáčknutí tabulátoru). Programovací jazyk python tak bude vědět, že takto odskočené příkazy souvisí s funkcí `while`. Neodskočené příkazy vynechá.

```
while True:  
    <kód>
```

6. Poté je potřeba vytvořit tři proměnné pro měření tří os akcelerometru ADXL345. Hodnota proměnných je funkce *Precti* z knihovny `kADXL345`. Tato funkce uloží do proměnné hodnotu zrychlení v jednotce `g`. Parametr `bus` značí sběrnici vytvořenou v bodě 3. `Adresa0` a `Adresa1` jsou adresy registrů pro jednotlivé osy zobrazené na obrázku 7 (jedna funkce *Precti* pro jednu osu).
7. Vytvořte proměnnou, jejíž hodnota je funkce *Zobraz* (knihovna `kADXL345`). Vstupní parametr `root` je proměnná odkazující na vytvořené zobrazovací okno (*pocatecní inicializace*). Další tři parametry jsou hodnoty naměřených zrychlení pro jednotlivé osy v jednotkách `g`. Tyto hodnoty je nutné zadat postupně (osa `x`, `y`, `z`)
8. Nakonec je ve funkci `while` nutné zavolat funkci *Konec\_mereni* (knihovna `kADXL345`). parametr `indikace` je proměnná s hodnotou funkce *Zobraz* (vytvořená v předchozím kroku). Parametr `okno` značí odkaz na vytvořené zobrazovací okno (bod 3.). Funkce slouží k ukončení programu při zavření zobrazovacího okna (ukončuje funkci `while`)

## 5 VIBRACE

### 5.1 Teorie k problematice vibrací

Vibrace je pohyb vlivem mechanických oscilací zařízení nebo součástky. Mohou být periodické, jako pohyb kyvadla nebo náhodné, například pohyb pneumatiky po šterku. Vibrace mohou být určeny pomocí metrických jednotek, což jsou  $\text{m/s}^2$  nebo v jednotkách gravitační konstanty  $g$ . Měřený objekt může vibrovat vlivem dvou druhů vibrací.

Jedním druhem jsou volné vibrace. Tyto vibrace vznikají, pokud je objekt vychýlen z rovnovážné polohy a je mu dovolena oscilace. Rezonance je snaha systému více oscilovat na určitých frekvencích než na frekvencích jiných. Pokud jsou blízko pozorovaného objektu vytvořené nucené vibrace, vlastní frekvence daného objektu způsobuje nárůst vibračních sil. Vlivem času tento nárůst frekvencí může způsobit velké vibrace (i při malé prvotní nucené vibraci) a může daný objekt poškodit.

Druhý druh vibrací jsou vibrace nucené. Tyto vibrace vznikají při působení měnící se síly. Rotační motor může vytvořit vibrace o hodnotách jiných, než je hodnota vlastní frekvence daného objektu. Vlivem, že objekt se skládá z více částí (kompresor, turbína) a každá část má jinou vlastní frekvenci, vzniká opotřebení. Tímto měřením lze určit opotřebení měřeného rotačního objektu (měření kondice objektu).[31]

Pro zjištění rychlosti otáčení rotačního tělesa, které tímto pohybem vyvolává vibrace lze určit pomocí naměřené velikosti frekvence. Rychlost otáčení zpravidla odpovídá první harmonické frekvenci frekvenčního spektra vibrací. Z této frekvence lze dostat rychlost otáčení v jednotkách RPM (otáčky za minutu) pomocí rovnice 10.

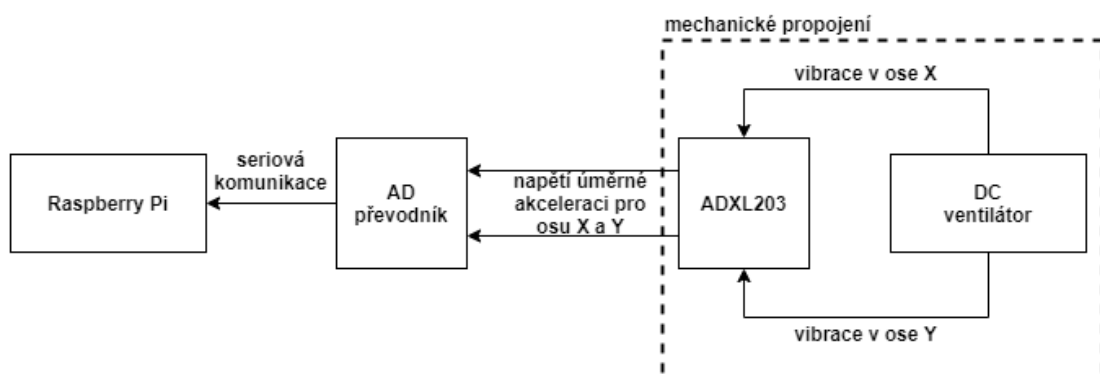
$$f_{RPM} = f_{Hz} * 60 \quad (10)$$

kde  $f_{RPM}$  je počet otáček za minutu a  $f_{Hz}$  je frekvence vibrací v jednotkách hertz.

Rovnice vychází z definice jednotky hertz, která udává počet opakujících se cyklů za jednotku času, tedy sekundu. Z této definice vyplývá že převedením hodnoty z hertz (počet opakování za sekundu) na RPM (počet otáček za minutu), je nutné tuto hodnotu vydělit 60 sekundami (pro opačný převod vynásobit hodnotou 60, rovnice 10).

Při zkoumání frekvenčního spektra vibrací lze toto spektrum rozdělit do tří částí. První část určuje základní frekvenci, která je v úloze určená pro výpočet otáček za minutu DC ventilátoru. Druhá část je v rozsahu od první harmonické frekvence, až po desátou harmonickou frekvenci. Na těchto frekvencích se projevují mechanické závady ventilátoru (nevyváženost, nesouosost). Frekvence, jejichž hodnota je větší, než desátá harmonická frekvence značí závady způsobené ložisky ventilátoru. [32]

Blokové schéma pro měření vibrací je na obrázku 21. Vibrace způsobené otáčením lopatek DC ventilátoru se zjistí dynamickými změnami zrychlení. Změna se zaznamená pomocí dvou os akcelerometru ADXL203 připevněných na rámu ventilátoru. Analogový výstup obou akcelerometrů (napětí) je pomocí A/D převodníků převedeno na digitální hodnotu. Po převedení se digitální signál přivede do mikropočítače Raspberry Pi. Zde se zjištěné hodnoty zaznamenají a zpracují pomocí potřebných algoritmů (FFT) pro určení frekvence a amplitudy vzniklých vibrací.



Obrázek 21 Blokové schéma úlohy pro měření vibrací

## 5.2 Popis zapojení

Pro měření vibrací způsobené vnějším zdrojem byl zvolen akcelerometr ADXL203. Tento akcelerometr byl zvolen z důvodu jeho přesnosti (nízká hodnota šumu) a velikosti maximální možné frekvence, kterou dokáže zaznamenat bez zkreslení. Maximální možná frekvence (pokles signálu o 3 dB) snímání je 2,5 kHz a je nastavená pomocí hodnoty kondenzátoru, jenž se nachází mezi výstupním pinem osy akcelerometru a zemí. Ve schématu v příloze B.4 jsou to kondenzátory  $C_1$  pro osu X a  $C_2$  pro osu Y. Hodnota kapacity pro frekvenční rozsah byla vypočtena z rovnice číslo 11, kterou udává výrobce akcelerometru. [16]

$$f_{-3dB} = \frac{1}{2\pi R_{FILT} C_{(x,y)}}, \quad (10)$$

$$C_{(1;2)} = \frac{1}{2\pi R_{FILT} f_{-3dB}}, \quad (11)$$

kde  $f_{-3dB}$  je maximální frekvence, kterou akcelerometr zaznamená bez útlumu,  $R_{FILT}$  je hodnota vnitřního rezistoru na výstupu os akcelerometru a  $C_{(1;2)}$  je hodnota kapacity kondenzátoru, který se připojí na osu X nebo Y a společně s rezistorem  $R_{FILT}$  tak tvoří filtr.[16]

Při dosazení hodnot odporu a frekvence hodnotami, které udává výrobce, do rovnice 11 je vypočtena nejmenší hodnota kapacity, pro kondenzátor  $C_1$ ,  $C_2$ :

$$C_{(1;2)} = \frac{1}{2\pi \cdot 32 \cdot 10^3 \cdot 2,5 \cdot 10^3} = 1,989 \text{ nF}. \quad (12)$$

Pro dosažení maximálního frekvenčního rozsahu je nutné zvolit hodnotu kondenzátoru 2 nF. Výrobce také udává, že alespoň tato hodnota kapacity (popřípadě

větší) musí být na výstupních pinech akcelerometru ADXL203. Z důvodu tolerance hodnot kapacity kondenzátorů v provedení SMD byla zvolena hodnota kapacity 2,2 nF.

Výstupní analogový signál akcelerometru je převeden na digitální signál pomocí A/D převodníku MCP3202. Tento A/D převodník je popsán v kapitole 3.3. MCP3202 je napájen napětím o velikosti 5 V. Toto napětí je nutné pro dostupný vstupní rozsah 0 až 5 V. Tento rozsah je důležitý, jelikož je to rozsah výstupního signálu, jenž poskytuje akcelerometr ADXL203 na jeho výstupech (osy X a Y).

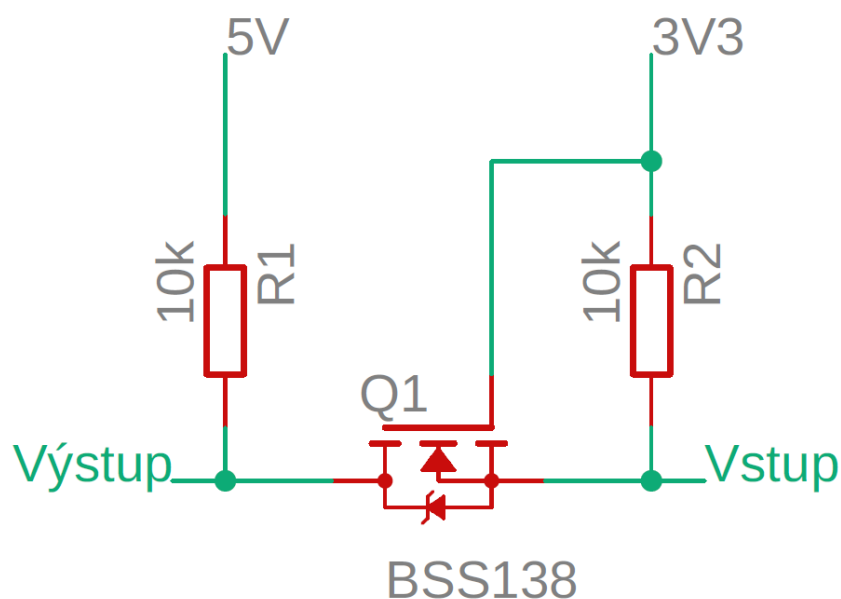
Z důvodu nerovnosti logických hodnot napětí na kanálech, jenž jsou určeny pro komunikaci, kde MCP3202 má při použití napájení sériovou komunikací v logice 5 V a mikropočítač Raspberry Pi pracuje se sériovou komunikací v logice 3,3 V, je nutné použít obvod pro přeměnu hodnot napětí. Tuto přeměnu zajišťuje obvod s názvem Level shifter, neboli posouvač napětí. Level shifter pro jeden kanál komunikace je znázorněn na obrázku 22. Obvod se skládá z jednoho rychlého FET tranzistoru typu BSS138 a dvou pull-up rezistorů. Každý z rezistorů je připojen na jinou hodnotu napájecího napětí. V tomto případě je jeden rezistor připojen na hodnotu napětí 5 V z výstupu A/D převodníku MCP3202 (MOSI, MISO, CE, SCLK). Druhý rezistor je připojen na napětí 3,3 V ze vstupu/výstupu GPIO pinu mikropočítače Raspberry Pi. Princip převodu napětí je poté následovný:

Pokud je pin s hodnotou napětí nižší (3,3 V) brán jako vstup a druhý pin (napětí 5 V) brán jako výstup pak:

1. je-li vstup v hodnotě logické 1, hodnota napětí přechodu mezi gate a source tranzistoru je nulová (stejný potenciál 3,3 V), tranzistor se vypne a pin vstupu je pomocí pull-up rezistoru nastaven na příslušné napětí značící logickou 1 (5 V);
2. je-li vstup nastaven na hodnotu logické 0 (teoretická hodnota 0 V), vzniká na přechodu gate a source napětí o velikosti příslušného napájecího napětí, tranzistor se otevře a stáhne napětí na drainu k hodnotě napětí, které je na source. Tím se dostává výstup do stejné logické úrovně jako vstup, tedy logická 0. Při této situaci dochází k tomu, že do vstupního pinu tečou proudy vzniklé z obou rezistorů. Zde je tedy nutné určit takovou hodnotu, aby se příslušný pin nezničil.

Pro druhý případ, kdy je nízké napětí (3,3 V) bráno jako výstup a vysoké napětí (5 V) bráno jako vstup, je práce level shifteru následující:

1. Je-li vstup v hodnotě logické 1, tranzistor je vypnutý a není žádný způsob, jako ho s pomocí vstupu sepnout. Z tohoto důvodu bude napětí na výstupu nastaveno pomocí pull-up rezistoru na hodnotu příslušného napětí, tedy také logické 1. Hodnota 3,3 V na výstupu (source) a gate tranzistoru zajišťuje jeho vypnutí, a ze vstupu tak na výstup neproudí žádný proud.
2. Je-li vstup v hodnotě logické 0, přes tranzistor začne proudit proud ze source na drain pomocí technologicky vyrobeného PN přechodu (technologické spojení source a substrát část P a drain jako část N). Tento proud poté stáhne hodnotu výstupu na hodnotu blízké (nebo rovno) logické 0. Hodnota napětí na source bude odpovídat hodnotě napětí na přechodu drain a source ( $V_{SD}$ ). Vzniklé napětí na source nesmí být velké. Rozdíl napětí mezi gate (napájecí napětí 3,3 V) a napětí na source ( $V_{SD}$ ) musí takové, aby otevřelo tranzistor. Po otevření tranzistoru dochází k nastavení logické 0 na výstupu a do vstupu poté proudí proud z obou rezistorů.



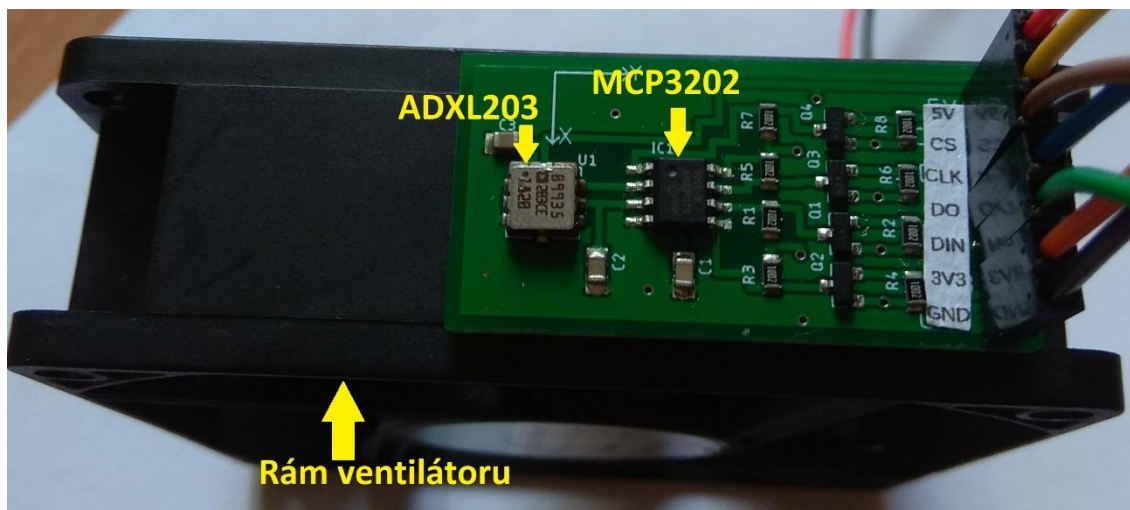
Obrázek 22 Schéma zapojení level shifteru pro převod 5 V sběrnice na 3,3 V sběrnici

Schéma zapojení a layout desky, vytvořené v návrhovém programu EAGLE, jsou zobrazeny v příloze B. Vstupní a výstupní datové piny, piny pro napájení 3,3 a 5 voltů a zem jsou z desky vyvedeny pomocí kabelů a lze přímo je propojit s GPIO piny, které poskytuje mikropočítač Raspberry Pi. Na obrázku 23 je vidět vytvořená deska připevněna na rámu DC ventilátoru pomocí oboustranně lepícího pásku. Značení kabelů je vypsáno v tabulce 1.

Tabulka 1 značení kabelů desky s akcelerometrem ADXL203

Značení na desce	Účel	Značení na Raspberry Pi	Barva
5V	Napětí 5 V	5V	Červená
CS	Zahájení komunikace	CE0	Žlutá
CLK	Hodinový signál	SCLK	Hnědá
DO	Výstup A/D převodníku	MISO	Modrá
DIN	Vstup A/D převodníku	MOSI	Zelená
3V3	Napětí 3,3 V	3V3	Oranžová
GND	Společná zem	GND	Černá





Obrázek 23 Upevnění desky s akcelerometrem ADXL203 na rámu DC ventilátoru

## 5.3 Zvolené knihovny

Pro sestrojení programu bylo zvolených pět knihoven. Jedná se o knihovny spidev, pyqtgraph, numpy, scipy a time.

### 5.3.1 Spidev

Knihovna spidev byla zvolena pro zajištění komunikace mezi A/D převodníkem a mikropočítačem. Tato komunikace probíhá pomocí sériového protokolu SPI. Knihovna pracuje s GPIO porty mikropočítače Raspberry Pi, jenž jsou zobrazené na obrázku 12 (kapitola 3. 4. 1.). Mezi tyto porty patří MISO, zajišťující komunikaci z A/D převodníku do mikropočítače, MOSI pro komunikaci z mikropočítače do A/D převodníku, SCLK určený pro hodinový signál, který zajišťuje správnou komunikaci a port CE0, který zajišťuje určení, s kterou součástkou se má komunikovat.

```
spidev.open(0, 0)
```

byla určena komunikaci se součástkou umístěnou na pinu CE0, tedy A/D převodníkem. (Raspberry Pi také poskytuje dvě možnosti připojení součástek, tedy také CE1). Pro určení, v jakém nastavení má komunikace pracovat slouží funkce

```
spidev.mode = 0b00
```

Nastavení *0b00* zajišťuje aktivní hodnotu hodinového signálu v úrovni logické 1 a zápis/čtení probíhá při nástupné hraně hodinového signálu. Nastavení může být v dalších třech módech, které způsobí různou kombinaci nastavených prvků.

```
spidev.xfer2(parametry)
```

Tato funkce provádí zápis hodnot, určených v parametrech funkce, do součástky, umístěné na dříve zvoleném pinu (CE0 nebo CE1). Tyto hodnoty jsou posílány přes pin MOSI. Ve stejnou dobu probíhá čtení ze součástky do proměnné v mikropočítači, která byla určena programově. Pokud součástka při zaslání potřebných hodnot nereaguje na další přicházející hodnoty určené k zápisu (zpravidla posílání hodnoty 0), dá se poté nastavit výčet většího množství informací, než bylo užitečně zapsáno. Při zapisování a čtení více osmibitových čísel funkce xfer2 nemění logickou hodnotu na portu CE a ten

zůstává konstantní. Tato možnost je klíčová pro komunikaci s periferiemi, které ji pro její správnost potřebují.

### 5.3.2 Pyqtgraph

Pro správné a přehledné měření byla využita knihovna pyqtgraph. Tato knihovna je napsaná čistě v programovacím jazyce python a umožňuje ovládat grafické rozhraní pro tvorbu 2D a 3D struktur. [33] Pyqtgraph je postaven na sadě nástrojů, které zajišťují přístup k Qt. Qt je multiplatformní prostředí pro vytváření aplikací na různých operačních systémech, a to od embedded systémů, až po operační systémy pro mobilní telefony. [34] Sady nástrojů, jenž pracují v prostředí Qt a jsou využívány knihovnou pyqtgraph (nutné pro správnou funkci knihovny) jsou PyQt4 nebo PySide a jejich novější verze. Dalším nutností pro použití pyqtgraph je knihovna numpy, kterou používá pro zpracování dat a pracuje s proměnnými, které tato knihovna poskytuje. [33]

Obnova grafu pro stálé vykreslování je provedena pomocí funkce *QTimer* a nastavení potřebných parametrů

```
timer = pg.QtCore.QTimer()
timer.timeout.connect(update)
timer.start(0)
```

První řádek funkce slouží k vytvoření proměnné odpovídající časovači. Tento časovač pochází z knihovny pyqtgraph (importována jako pg) a umožňuje neustálé vykonávání později určené funkce (podobně jako příkaz while). Následující řádek kódu slouží k určení, o jakou funkci se jedná. Zde je to funkce update, která je základem celého programu. Poslední řádek nastavení časovače určuje, s jakým zpožděním v jednotkách milisekund se má časovač spustit (0 ms pro spuštění, ihned).

Funkce *if*, jenž je napsaná na konci programu slouží ke spuštění grafického uživatelského rozhraní (funkce *QApplication.instance().exec\_()*), které je provozováno pomocí dříve nastaveného časovače, dokud nejsou ukončené veškeré widgety (okna), které jsou příslušnými funkcemi vytvořeny nebo není ukončen program.

```
if __name__ == '__main__':
    import sys
    if (sys.flags.interactive != 1) or not hasattr(QtCore,
'PYQT_VERSION'):
        QtGui.QApplication.instance().exec_()
```

První *if* příkaz kontroluje, zda je tento soubor hlavní modul (nejedná se o knihovnu, není importovaný). Tato podmínka je splněná, jelikož je tento soubor vytvořený pro vykonání programu, nikoli jako knihovna.

Následuje importování knihovny *sys*, která slouží pro přístup k proměnným a funkcím interpreta (počítačový program, který vykonává instrukce). Tyto proměnné a funkce slouží k například k určení nastavení aktuálního systému (přítomnost flagů). Z této knihovny je využita funkce *sys.flags.interactive*. Tato funkce určuje, zda je program v interaktivním módu. [35] Interaktivní mód je určitý druh souboru, jenž po napsání řádku kódu poskytuje ihned jeho vykonání (většinou psaný v terminále). Tento výsledek je poté určen pomocí dříve napsaných řádků. Interaktivní mód je tak opakem módu skript, který využívá napsání celého programu v určitém souboru a až poté provádí jeho vykonání. [36]

Funkce `hasattr(QtCore, 'PYQT_VERSION')` kontroluje, zda je nainstalované sady nástrojů PyQt a kontroluje jejich verzi.[37] Pokud by tyto sady nebyly nainstalované funkce pro spuštění grafického rozhraní by se nespustila. Novější verze podporují kromě PyQt také sadu nástrojů PySide. Při použití této sady je nutné tuto podmínku odstranit, aby mohlo být spuštěné grafické rozhraní.

Knihovna `pyqtgraph` byla zvolena převážně pro její rychlost, kterou poskytuje. Tato rychlost je nutná pro zaznamenání a zobrazení signálů v reálném čase, které měří akcelerometr.

### 5.3.3 Numpy a Scipy

Pro složitější výpočty a přístup k funkcím potřebných k analýze fft byly využity knihovny `numpy` a `scipy`. Obě knihovny slouží pro usnadnění výpočtů v oblasti matematiky, vědy a techniky. Knihovna `numpy` zajišťuje práci s N-rozměrnými poli, zajišťuje možnost využití konstant (například  $\pi$ ) a využití lineární algebry a práce s náhodnými čísly. [38] Použití knihovny `numpy` je nutné také z důvodu knihovny `pyqtgraph`, která tuto knihovnu využívá. Knihovna `scipy` rozvíjí knihovnu `numpy` a některé další knihovny a poskytuje tak uživatelsky přijatelné a efektivní výpočty

```
scipy.fftpack.fft(parametry)
```

Funkce provede algoritmus FFT (kapitola 6.5) na naměřené vzorky, které se do funkce předají nejčastěji jako parametr typu pole. Návrátová hodnota této funkce je pole komplexních čísel, které má jako první prvek (prvek 0) hodnotu velikost nulové frekvence (stejnosečná složka). Hodnoty od druhého prvku (prvek 1) až po prvek, jehož hodnota je polovina všech nasnímaných vzorků ( $N/2$ ), představuje kladné frekvence. Zbylé hodnot představují záporné frekvence.

```
numpy.fft.fftfreq(N, T)
```

Tato funkce na základě předaných parametrů vytvoří pole typu list. Hodnoty v tomto listu poté odpovídají hodnotám jednotlivých frekvencí, které jsou kompatibilní s hodnotami modulů, které vypočítá funkce `fft()`. První polovina opět uchovává frekvence kladné a druhá polovina tohoto pole určuje frekvence záporné. Parametry, které se této funkci poskytují jsou počet nasnímaných vzorků ( $N$ ) a perioda ( $T$ ), se kterou byly vzorky nasnímány. [39]

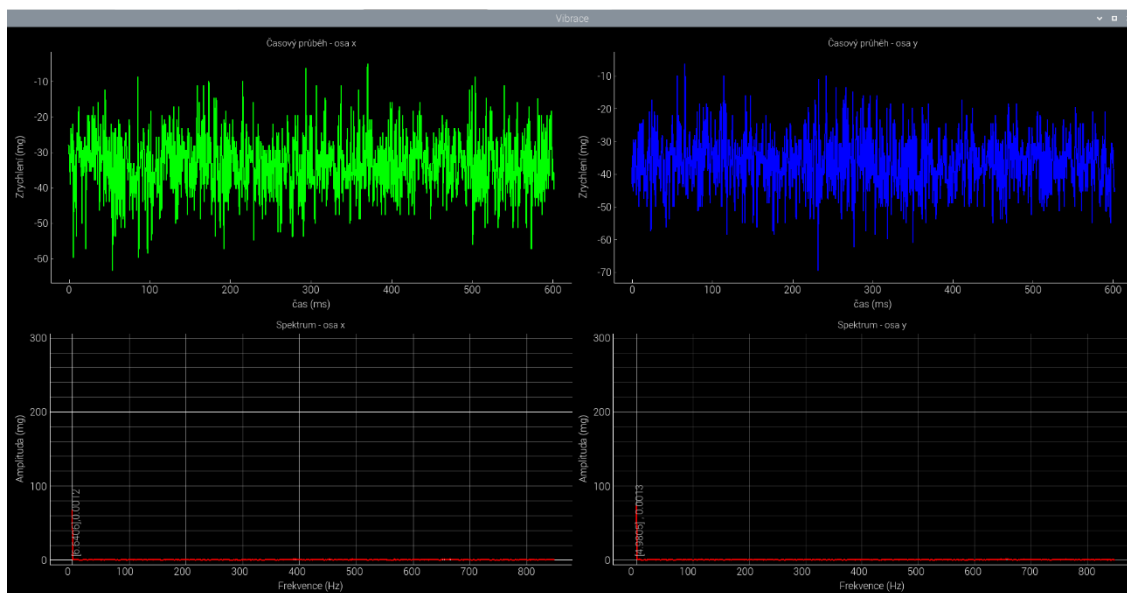
## 5.4 Princip měření

Základem měření je nasnímání určitého množství vzorků signálu a následně tyto vzorky zpracovat. Snímaný signál je vytvořen vibracemi, které působí na akcelerometr ADXL203. Ten dokáže tyto vibrace zachytit v podobě změny hodnoty zrychlení. Signál z akcelerometru je pomocí A/D převodníku MCP3202 vzorkován a přeměněn na digitální signál, který je možný zaznamenat pomocí sériové komunikace SPI na mikropočítači Raspberry Pi. Takto získané vzorky je poté možno ihned zobrazit pomocí grafického rozhraní. Pokud se vzorky vynesou do grafu společně s časem, vzniká časový průběh působících vibrací.

Pro určení frekvenčního spektra a zjištění dominantní frekvence a amplitudu vibrací je nutné vzorky zpracovat. Zpracování je provedeno pomocí analýzy rychlé Fourierovy transformace (FFT). Tato analýza je provedena pomocí funkce `fft()`, která je součástí

knihovny scipy (kapitola 5.3.3). Pomocí funkce `fftfreq()` bylo určené frekvenční pole příslušné k analýze FFT.

Takto naměřené a vypočtené hodnoty jsou zobrazeny pomocí knihovny `pyqtgraph` do grafů. Zobrazovací okno s grafy je vidět na obrázku 24. Snímání vzorků probíhá po blocích o velikostech počtu vzorků  $N$ . Blokové snímání je použito z důvodu lépe nastavitelné časové mezery mezi jednotlivými vzorky (vzorkovací frekvence), snímání není zpomalováno obnovou grafického rozhraní, které má větší časovou náročnost, která by ovlivňovala frekvenci snímání a následné použití funkce `fft` na tento blok (pole nasnímaných vzorků). Frekvence je při spuštění programu naměřena a spočítána. Vyčítání vzorků je softwarově zpomaleno pro zajištění vzorkovací frekvence pod úrovní frekvenčního rozsahu akcelerometru ADXL203 (pod hodnotou 2 kHz). Vzorky jsou tedy vyčteny nejdříve pro osu X a až poté pro osu Y ve dvou funkcích typu `for`. Kvůli této neplynulosti, kdy vyčtení vzorku a jeho následné zobrazení neprobíhá postupně, ale po blocích, nastává při zobrazení, obnovování a práci s oknem grafů určitá časová prodleva. Při vyčtení většího počtu vzorků může tato časová prodleva dosahovat velikosti v řádech stovek milisekund a více. Tento jev se poté projeví na zobrazování grafu a jejich obnovení se nebude zdát plynulé. Zrychlení obnovy zaznamenání grafu a dosažení plynulosti grafů, tak může být dosaženo snížením počtu nasnímaných vzorků. Toto snížení ovšem také snižuje rozlišení výsledných grafů. Odečtení hodnot z grafů frekvenčního spektra lze pomocí ukazatele *peak*, který zobrazuje frekvenci s nejvyšší amplitudou. Druhým ukazatelem je ukazatel *kurzor*, který ukazuje hodnotu po zmačknutí levého tlačítka myši v poli grafu. Po zmačknutí nastává určitá časová prodleva mezi stiskem tlačítka a zobrazením hodnoty. Tato prodleva je důvodem delší obnovy grafu (blokové měření) a než je stisk zaznamenaný, jeho indikace trvá tři až čtyři snímací cykly.



Obrázek 24 Vytvořené okno pro zobrazení grafů s naměřenými hodnotami

## 5.5 Rychlá Fourierova transformace – FFT

Rychlá Fourierova transformace představuje jeden z nejrozšířenějších a nejrychlejších

algoritmů pro získání frekvenčního spektra ze vzorkovaného signálu. Jedná se zlepšení algoritmu diskrétní Fourierově transformaci (DFT). DFT slouží k přeměně  $N$  počtu dat, které jsou od sebe vzdálené o stejnou velikost  $T$  (vzorkovaný signál) na posloupnost o stejném počtu dat (například jednotlivé frekvence a jejich amplitudy). Jedná se tedy o přepočítání z časové charakteristiky do frekvenční charakteristiky. [40] Pro výpočet DFT lze použít vzorec 12.

$$X[k] = \sum_{n=0}^{N-1} x(n) * e^{\frac{-j*2*\pi*n*k}{N}} \quad \text{pro: } k = 0, 1, \dots, N-1, \quad (13)$$

kde  $X[k]$  představuje hodnotu magnitudy frekvence  $k$  o příslušné velikosti,  $N$  je celkový počet nasnímaných vzorků,  $x(n)$  je hodnota příslušného vzorku  $n$ . [41]

Analýza FFT využívá tzv. motýlků. Tento pojem vychází z grafického znázornění signálových toků při počítání FFT. Urychlení nastává ve výpočtu dvou menších DFT o velikosti  $N/2$ . Na tyto vypočtené DFT je poté proveden algoritmus motýlek pro vypočtení celkové posloupnosti Fourierovy řady. Před vypočítáním DFT je nejdříve nutné vzorkovaný signál rozdělit na sudé a liché vzorky (bitově-reverzované pořadí). [40]

Časová náročnost výpočtu DFT je závislá na počtu vzorků  $N$ , se kterými se počítá. Časová náročnost vzniká především při operaci násobení, jelikož je tato operace náročnější než sčítání. Celková náročnost algoritmu DFT je poté rovna kvadrantu této hodnoty, tedy  $N^2$ . Počet stupňů potřebných na vypočtení FFT je dán vztahem:

$$m = \log_2 N, \quad (14)$$

kde  $m$  představuje počet potřebných stupňů a  $N$  představuje počet vzorků. [40]

Každý stupeň algoritmu FFT obsahuje  $N/2$  počet motýlků, jenž je nutné spočítat. Každý motýlek poté obsahuje průměrně jednu operaci násobení, která díky své náročnosti vytváří časovou náročnost. Celková časová náročnost algoritmu FFT se rovná:

$$O\left(\frac{N}{2} * m\right) = O\left(\frac{N}{2} * \log_2 N\right), \quad (15)$$

kde  $O()$  představuje časovou náročnost algoritmu,  $N$  je počet vzorků a  $m$  je počet stupňů algoritmu FFT. [40]

Tabulka 2: Porovnání rychlosti mezi algoritmem DFT a FFT

N	Rychlost vzorku [ns]	O(DFT)	Celkový čas DFT	O(FFT)	Celkový čas FFT
1024	1,0	$1,048 * 10^6$	1,048 ms	5120	5,120 $\mu$ s
$10^6$		$10^{12}$	16,66 min	$9,966 * 10^6$	9,966 ms
$10^9$		$10^{18}$	31,7 let	$14,95 * 10^9$	14,95 s

Z tabulky 2 je vidět, že při zpracování jednoho vzorku s rychlostí 1 ns, lze při zpracování  $10^9$  počtu vzorků, dosáhnout celkového času zpracování signálu pomocí algoritmu FFT 14,95 s. Stejný počet vzorků se stejnou rychlostí jejich zpracování by algoritmem DFT trval 31,7 let.

## 6 MĚŘENÍ

Měření je rozděleno do dvou částí, a to měření statistických parametrů vůči gravitační síle působící na akcelerometr ADXL345 pomocí vyrobeného aktuátoru (obrázek 19). Druhou část tvoří změření frekvenčního spektra vibrací, které způsobuje rotace lopatek ventilátoru, pomocí akcelerometru ADXL203 a A/D převodníku MCP3202.

### 6.1 Měření s akcelerometrem ADXL345

Akcelerometr je upevněn na aktuátor, zajišťující změnu jeho náklonu a rotace vůči působení gravitační síly. Dále jsou jeho piny připojeny na GPIO piny mikropočítače Raspberry Pi podle kapitoly 4.2.2. Na mikropočítači je v programovacím prostředí pro jazyk Python (například Thonny Python IDE) napsán program pro práci s akcelerometrem ADXL345 (kapitola 4.4.1). Se správně napsaným programem a připojeným akcelerometrem je možné provést měření následujících parametrů.

Při spuštění programu musí být akcelerometr v poloze  $0^\circ$  pro náklon i rotaci. Důvodem je jeho inicializace a vypočtení správné hodnoty offset, která se zapíše do příslušných registrů a vyrovnává tak naměřené hodnoty vůči této prvotní inicializaci. Z hodnoty zobrazené na mikropočítači je ihned po inicializaci možné zjistit hodnotu offsetu (posunutí nuly), kterou akcelerometr nedokáže vykompenzovat. Maximální hodnota offsetu by se tak měla rovnat hodně LSB, která představuje nevykompenzování změřené akcelerace. Tuto hodnotu udává výrobce jako 3,9 mg/LSB. Důvodem tohoto offsetu je zaokrouhlování při vypočítání hodnoty, která se zapisuje do offset registru a slouží pro kompenzaci. Toto zaokrouhlování tak může způsobit samotný offset. Pro vypočítání maximální hodnoty offsetu se do rovnice 6 dosadí příslušná hodnota akcelerace ( $\pm 3,9$  mg). Pro zrychlení v ose z je nutné k této akceleraci přičíst hodnotu, značící působení gravitační síly v této ose o velikosti 1 g (tedy 0,9961 a 1,0039 mg).

$$\tan \theta_{xyz} = \frac{-0,0039}{\sqrt{0,0039^2 + 1,0039^2}} * \frac{180}{\pi} = -0,2225, \quad (16)$$

$$\theta_{xyz} = \tan^{-1}(-0,2256) = -0,22^\circ. \quad (17)$$

Při odchýlení v druhém směru bude platit

$$\tan \theta_{xyz} = \frac{-(-0,0039)}{\sqrt{(-0,0039)^2 + 0,9961^2}} * \frac{180}{\pi} = -0,2243, \quad (18)$$

$$\theta_{xyz} = \tan^{-1}(-0,2301) = 0,22^\circ. \quad (19)$$

Pokud jsou hodnoty náklonu a rotace po počáteční inicializaci v rozmezí  $-0,22^\circ$  až  $0,22^\circ$  hodnota offsetu je v normě, kterou udává výrobce.

Rozsah měření je další parametr, který se dá určit jako nejmenší a největší možná naměřená hodnota. Pro měření tohoto parametru měním náklon (rotaci) akcelerometru pomocí aktuátoru, až do doby, kdy se začíná měřená hodnota zvětšovat (pro měření

minimální možné hodnoty) nebo zmenšovat (pro měření maximální možné hodnoty.

Po změření rozsahu je možné tento rozsah proměřit. Z těchto naměřených hodnot je možné získat citlivost senzoru a korekční křivku senzoru.

Pro měření citlivosti  $K$  senzoru tedy změny jeho výstupu v závislosti na změně vstupu lze určit z naměřených hodnot, popřípadě grafů, jako podíl rozdílu hodnot  $y$  a hodnot  $x$  pro dva body charakteristiky (sklon charakteristiky), tedy:

$$K = \frac{\Delta y}{\Delta x}, \quad (20)$$

kde  $K$  je citlivost senzoru,  $\Delta y$  je rozdíl výstupních hodnot senzoru a  $\Delta x$  je rozdíl vstupních hodnot senzoru

K určení rozsahu, ve kterém akcelerometr pracuje správně je nutné určit toleranční pásmo. Výrobce akcelerometru udává hodnotu 256/g. Tato hodnota značí že při změně výstupní hodnota při změřeném zrychlení 1 g je rovna číslu 256. Pro nejmenší možnou změnu, kterou akcelerometr zaznamená (změna v LSB) je tedy nutné hodnotu 1 g vydělit příslušným číslem 256. Pro hodnotu změny LSB poté platí:

$$LSB = \frac{a_{1g}}{\text{hodnota jednoho } g} = \frac{1}{256} = 3,9 \text{ mg}. \quad (21)$$

Tato hodnota udává nejmenší zachytitelnou změnu vstupu, která změní hodnotu na výstupu o jedna. Z důvodu stejné akcelerace, jako při počítání rozsahu přijatelného offsetu, má toleranční pásmo absolutní chyby stejnou velikost (rovnice 17 a 19). Naměřené absolutní chyby by neměly překročit hranici  $\pm 0,22^\circ$ .

Korekční křivka senzoru značí absolutní chybu naměřené hodnoty od reálné. Jedná se tak o hodnoty, které je nutné přičíst (odečíst) od naměřené hodnoty, aby se snížila chyba měření. Hodnotu absolutní hodnoty lze zjistit jako:

$$\Delta\theta = \theta_{\text{nastavená}} - \theta_{\text{změřená}}, \quad (22)$$

Kde  $\Delta\theta$  je absolutní chyba měření,  $\theta_{\text{nastavená}}$  je nastavená hodnota na rotačním stolku (vstupní hodnota) a  $\theta_{\text{změřená}}$  je hodnota akcelerometru (výstupní hodnota).

Při měření s akcelerometrem ADXL345 a příslušným aktuátorem je změna parametru úhlu rotace zajištěna pomocí rotačního stolku 1 a změna úhlu náklonu je zajištěna pomocí rotačního stolku 2 (značení na obrázku 19). Měření bylo provedeno nejdříve pro změnu rotace nastavením příslušného úhlu rotačním stolem 1 při konstantní poloze rotačního stolku 2 na hodnotě  $0^\circ$ . Pro měření náklonu změna probíhala opačně (konstantní poloha rotačního stolku 1 v poloze  $0^\circ$ ). Maximální a minimální naměřené hodnoty společně s hodnotami určené k proměření daného rozsahu jsou zapsány v tabulce 3. V tabulce ne nachází sloupec značený  $\Delta\theta$ . Jedná se o absolutní chybu mezi nastavenou hodnotou a změřenou. Tato hodnota se tedy vypočítá pomocí vzorce 22 a pro první řádek tabulky 3 a rotaci platí:

$$\Delta\theta = \theta_{\text{nastavená}} - \theta_{\text{změřená}} = (-91,67) - (-88,66) = -3,01. \quad (23)$$

Tabulka 3 Naměřené hodnoty úhlů rotace a náklonu akcelerometrem ADXL345

Rotace			Náklon		
$\theta_{\text{nastavený}} [^\circ]$	$\theta_{\text{změřený}} [^\circ]$	$\Delta\theta [^\circ]$	$\theta_{\text{nastavený}} [^\circ]$	$\theta_{\text{změřený}} [^\circ]$	$\Delta\theta [^\circ]$
-91,67	-88,66	-3,01	-91,67	-89,03	-2,63
-90,00	-87,89	-2,11	-90,00	-88,19	-1,81
-80,00	-78,85	-1,15	-80,00	-78,87	-1,13
-70,00	-69,07	-0,93	-70,00	-68,93	-1,07
-60,00	-59,12	-0,88	-60,00	-59,47	-0,53
-50,00	-49,63	-0,37	-50,00	-49,53	-0,47
-40,00	-39,73	-0,27	-40,00	-39,56	-0,44
-30,00	-29,88	-0,12	-30,00	-29,82	-0,18
-20,00	-19,86	-0,14	-20,00	-19,78	-0,22
-10,00	-9,94	-0,06	-10,00	-9,94	-0,06
0,00	0,00	0,00	0,00	0,22	-0,22
10,00	9,94	0,06	10,00	10,12	-0,12
20,00	19,79	0,21	20,00	20,13	-0,13
30,00	29,78	0,22	30,00	29,96	0,04
40,00	39,74	0,26	40,00	39,80	0,20
50,00	49,47	0,53	50,00	49,88	0,12
60,00	59,15	0,85	60,00	59,42	0,58
70,00	69,19	0,81	70,00	69,43	0,57
80,00	78,64	1,36	80,00	79,03	0,97
90,00	88,35	1,65	90,00	88,64	1,36
92,00	89,68	2,32	91,00	89,77	1,23

Měřicí rozsah nastavené hodnoty akcelerometru ADXL345 pro rotaci je  $-91,67^\circ$  až  $92^\circ$  a pro náklon byl změřen rozsah  $-91,67^\circ$  až  $91^\circ$ . I když tyto hodnoty překračují hodnotu  $90^\circ$  změřená hodnota akcelerometru nedosahuje hodnot  $90^\circ$  ani v jednom krajním bodu a největší absolutní chyba v těchto bodech dosahuje hodnoty  $-3,01^\circ$ . Tyto absolutní chyby a fakt, že akcelerometr nedokáže změřit hodnotu  $90^\circ$  (popřípadě  $-90^\circ$ ) je způsoben změnou jeho citlivosti na LSB. Z důvodu zpracování dat pouze s konstantní hodnotou citlivosti na LSB nastává při krajních hodnotách chyba měření.

Na obrázku 25 je znázorněna závislost nastavené hodnoty na rotačním stolku 1 na naměřené hodnotě akcelerometrem ADXL345. Citlivost akcelerometru ADXL345 je rovna sklonu křivky na obrázku 25. Citlivost na změnu úhlu rotace  $K$  lze vypočítat pomocí rovnice 20 a je spočítaná pro kladnou i zápornou část charakteristiky:

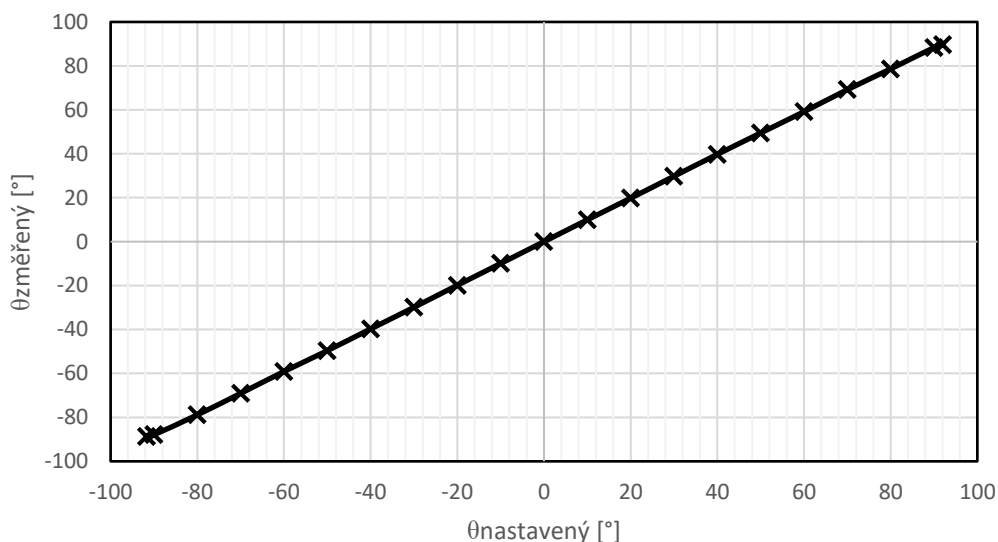
$$K = \frac{\Delta y}{\Delta x} = \frac{29,77 - 19,79}{30 - 20} = 0,998. \quad (24)$$

Pro zápornou část závislosti je hodnota citlivosti rovna:

$$K = \frac{\Delta y}{\Delta x} = \frac{-19,86 - (-29,88)}{-20 - (-30)} = 1,002. \quad (25)$$



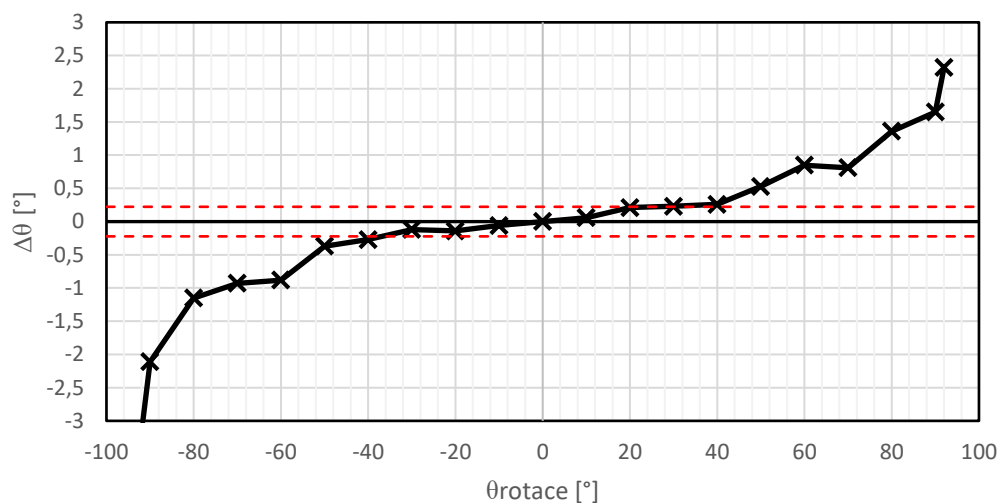
Graf závislosti nastavené hodnoty rotace na změřené hodnotě



Obrázek 25 Graf závislosti nastavené hodnoty rotace na změřené hodnotě

Z korekční křivky, která je vyobrazena na obrázku 26 lze určit měřicí rozsah, ve kterém akcelerometr ADXL345 pracuje v tolerančním pásmu, které určuje výrobce. Toto pásmo bylo vypočítané pomocí hodnoty zrychlení, která je přidělena nejmenšímu bitu měřicího registru. Z této hodnoty byl vypočten rozsah, kterému by měla náležet naměřená hodnota absolutní odchylky  $\Delta\theta$ , tedy interval  $-0,22^\circ$  až  $0,22^\circ$  (rovnice 17 a 19). Na obrázku 26 je toto pásmo znázorněno červenými čárkovanými čarami. Z těchto parametru lze určit, že akcelerometr ADXL345 pracuje správně v rozsahu  $-30^\circ$  až  $30^\circ$  při měření rotace. Z obrázku 26 lze vyčíst, že čím je nastavená hodnota vzdálenější od rozsahu správného měření, tím se absolutní chyba zvětšuje. To je zapříčiněno měnící se hodnotě zrychlení odpovídající nejmenšímu bitu měřicího registru (LSB). Tato hodnota je závislá na pozici akcelerometru v prostoru a se zvyšujícím se úhlu tato hodnota klesá. Při překročení určité hodnoty je tedy hodnota *LSB* (rovnice 21) je nedostačující a vzniká nárůst absolutní chyby.

Graf korekční křivky pro měření rotace



Obrázek 26 Graf korekční křivky pro měření rotace

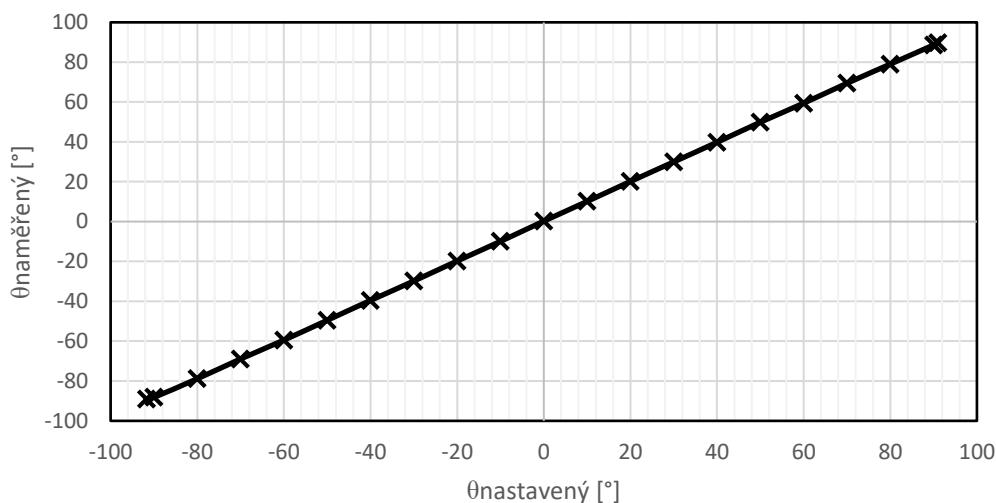
Pro náklon platí stejná pravidla, jako pro rotaci. Závislost naměřené hodnoty na nastavené je zobrazena na obrázku 27. Citlivost akcelerometru na změnu náklonu se vypočítá stejným způsobem jako pro rotaci (rovnice 20), tedy:

$$K = \frac{\Delta y}{\Delta x} = \frac{29,96 - 20,13}{30 - 20} = 0,983, \quad (26)$$

a pro záporné hodnoty závislosti:

$$K = \frac{\Delta y}{\Delta x} = \frac{-19,78 - (-29,82)}{-20 - (-30)} = 1,004. \quad (27)$$

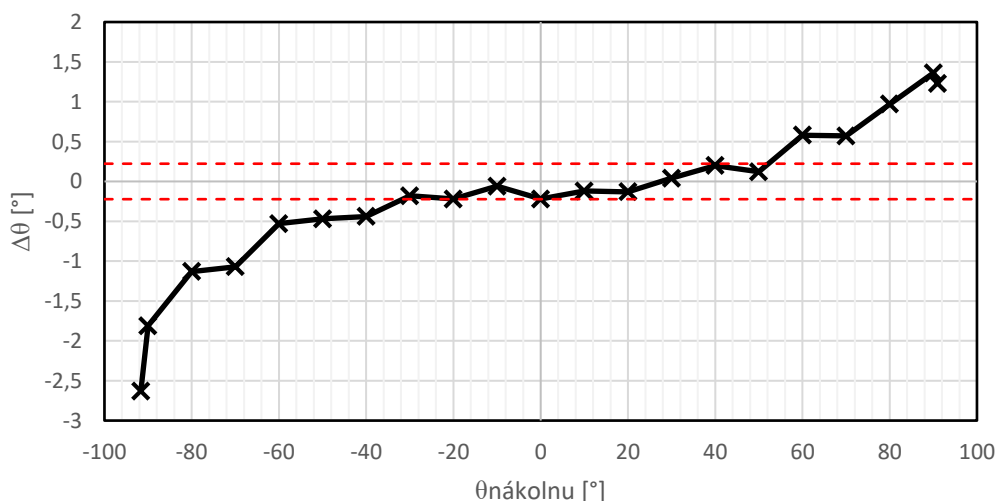
Graf závislosti nastavené hodnoty náklonu na změřené hodnotě



Obrázek 27 Graf závislosti nastavené hodnoty náklonu na změřené hodnotě

Korekční křivka náklonu, která je zobrazena na obrázku 28 určuje měřicí rozsah, ve kterém akcelerometr pracuje v tolerančním pásmu a jeho absolutní chyba není větší než limitující hodnota. Interval tolerančního pásma je stejný jako pro rotaci, tedy  $-0,22^\circ$  až  $0,22^\circ$ . Měřicí rozsah, ve kterém akcelerometr měří správnou hodnotu náklonu je  $-30^\circ$  až  $50^\circ$ . Absolutní chyba se při zvyšování hodnoty nad hranici měřicího rozsahu zvyšuje. Důvod je stejný jako při měření rotace, tedy změna hodnoty zrychlení připadající na nejmenší bit měřicího registru (LSB).

Graf korekční křivky pro měření náklonu



Obrázek 28 Graf korekční křivky pro měření náklonu

## 6.2 Měření s akcelerometrem ADXL203

Deska s akcelerometrem ADXL203 je připevněn na rámu DC ventilátoru a propojen s mikropočítačem Raspberry PI (kapitola 6.2). Ventilátor pro generaci vibrací byl zvolen EE80251B1-000U-A99 od výrobce Sunon. Pracovní otáčky ventilátoru uvádí výrobce 3200 RPM při pracovním napětí 12 V. [42] Pomocí programu mereni\_vibraci.py se spustí komunikace s akcelerometrem a získané hodnoty se vynesou do grafů. Po spuštění programu je možné změřit frekvenci, která je vybudena otáčkami akcelerometru. Tyto frekvence je možné odečíst pomocí kurzoru přímo z grafu. Osa X akcelerometru ADXL203 odpovídá vibracím ve směru osy otáčení lopatek ventilátoru. Na ose Y jsou měřeny vibrace kolmé na osu otáčení.

Z naměřených hodnot a grafů lze určit základní frekvenci. Z této frekvence lze určit velikost otáček ventilátoru v jednotkách RPM (otáčky za minutu). Z vyšších harmonických frekvencí lze analyzovat závady, které mohou nastat vlivem nedokonalosti ložisek ventilátoru.

Vibrace ventilátoru byly měřeny od hodnoty 4,5 V (nejmenší hodnota napětí, kterou udává výrobce) až po hodnotu 12 V (pracovní hodnota napětí DC ventilátoru). Hodnoty byly zvoleny 4,5 V, 6 V, 8 V, 10 V, 12 V. Při všech hodnotách se z grafů odečetla hodnota první harmonické a dalších třech dominantních frekvencí frekvenčního spektra způsobených vibrací. Naměřené hodnoty těchto vibrací jsou vyčteny v tabulce 4.

Tabulka 4 Naměřené hodnoty frekvenčního spektra pro měřené hodnoty napětí

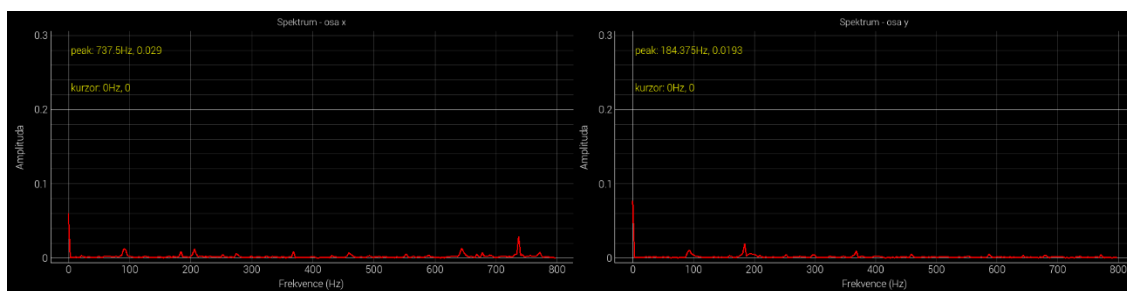
Vibrace v ose X			Vibrace v ose Y		
f <sub>Hz</sub> [Hz]	Amplituda [-]	f <sub>RPM</sub> [RPM]	f <sub>Hz</sub> [Hz]	Amplituda [-]	f <sub>RPM</sub> [RPM]
U = 4,5 V					
21,60	0,002	1296			
92,80	0,012		93,87	0,009	
205,70	0,012		184,38	0,019	
737,50	0,029		368,25	0,009	
U = 6 V					
31,25	0,101	1875	32,44	0,033	1946
60,25	0,067		61,11	0,023	
91,78	0,034		268,94	0,039	
272,84	0,044		300,00	0,051	
U = 8 V					
40,63	0,163	2401	40,63	0,059	2401
76,53	0,051		78,51	0,024	
116,19	0,040		109,23	0,028	
228,08	0,051		266,89	0,034	
U = 10 V					
46,01	0,054	2761	47,80	0,028	2868
181,29	0,036		137,50	0,037	
692,92	0,041		267,92	0,017	

728,13	0,099		696,78	0,025	
U = 12 V					
53,13	0,044	3188	52,92	0,042	3175
209,38	0,098		159,39	0,024	
259,61	0,063		262,50	0,054	
627,82	0,058		559,7	0,025	

Hodnota  $f_{RPM}$  pro lopatky DC ventilátoru se spočítá pomocí základní frekvence vibrací pomocí rovnice 10. RPM lopatek ventilátoru v ose X se pro napětí 4,5 V vypočítá jako:

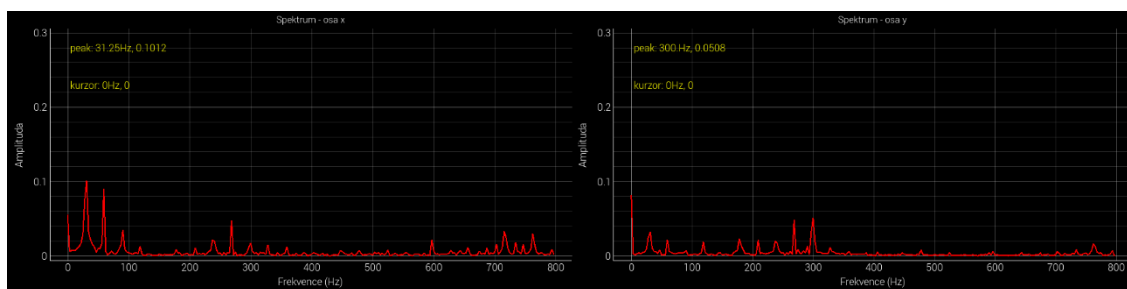
$$f_{RPM} = f_{Hz} * 60 = 21,60 * 60 = 1296 \text{ RPM} \quad (28)$$

Na obrázku 29 je zobrazeno naměřené frekvenční spektrum při napětí ventilátoru 4,5 V. Hodnota základní frekvence osy Y, která by měla být rovna hodnotě základní frekvence osy X, nebyla zachycena měřeným akcelerometrem. Ve frekvenčním spektru se poté projevují vyšší harmonické frekvence, které znázorňují vady ventilátoru.



Obrázek 29 Frekvenční spektrum vibrací ventilátoru při napájecím napětí 4,5 V

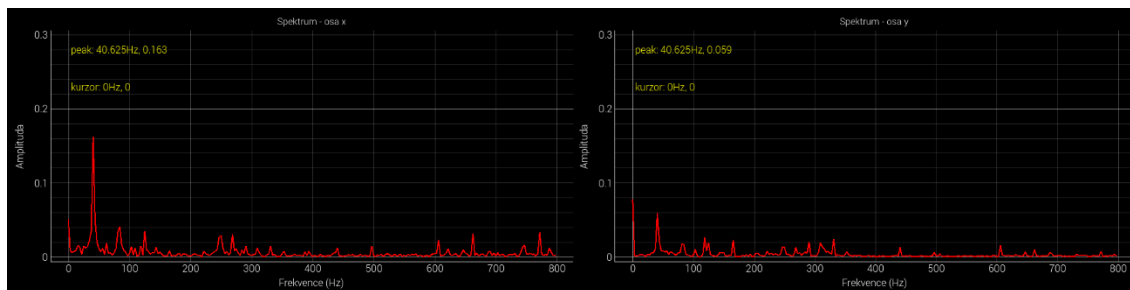
Na obrázku 30 je zobrazeno frekvenční spektrum při napájecím napětí ventilátoru 6 V. Zde je již možné měřit hodnotu základní frekvence vibrací pro osy X a Y. Z těchto frekvencí byly vypočteny otáčky za minutu lopatek ventilátoru, které se při tomto napájecím napětí odpovídají hodnotě 1900 RPM. Další dominantní frekvence pro osu x jsou 2., 3. a 9. harmonické základní frekvence. Pro osu Y jsou to frekvence 2., 9. a 10. harmonické.



Obrázek 30 Frekvenční spektrum vibrací ventilátoru při napájecím napětí 6 V

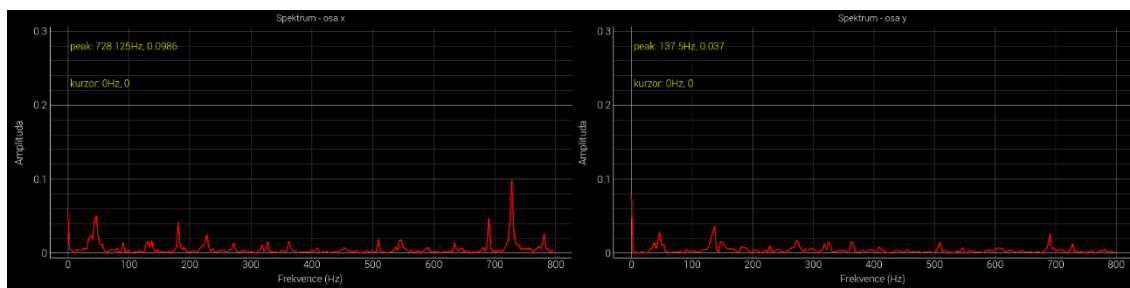
Na obrázku 31 je frekvenční spektrum při napájecím napětí 8 V. Zde jsou výrazné

hodnoty základních frekvencí. Ostatní harmonické násobky základních frekvencí nepůsobí dominantně jako v předchozích případech. Hodnota otáček z naměřených frekvencí je 2401 RPM.



Obrázek 31 Frekvenční spektrum vibrací ventilátoru při napájecím napětí 8 V

Frekvenční spektrum při napájecím napětí 10 V je zobrazeno na obrázku 32. Zde základní frekvence již nepůsobí dominantně. Velikost otáček zde odpovídá hodnotě 2761 RPM. Dominantní frekvence pro osu X odpovídá 16 harmonické frekvenci. Pro osu Y je to poté hodnota 3 harmonické.



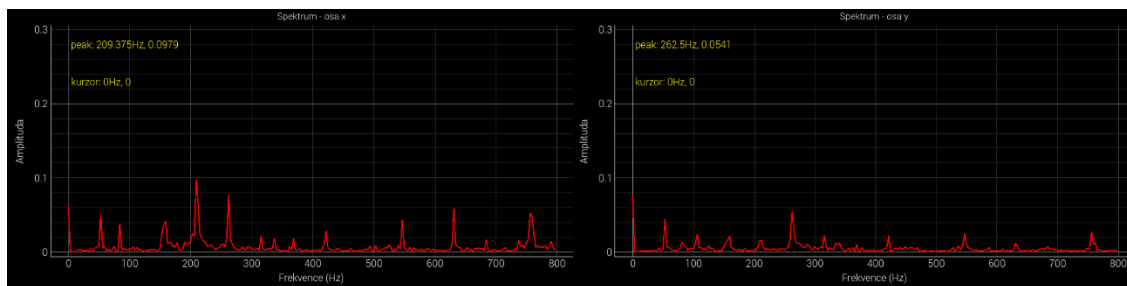
Obrázek 32 Frekvenční spektrum vibrací ventilátoru při napájecím napětí 10 V

Na obrázku 33 je zobrazeno frekvenční spektrum při napájecím napětí 12 V. Toto napětí je pracovní napětí DC ventilátoru, které udává jeho výrobce. V katalogovém listu DC ventilátoru výrobce udává velikost otáček při pracovním napětí  $3200 \text{ RPM} \pm 10 \%$ . Rozsah přijatelných otáček tak interval od 2880 do 3520 RPM. Tento rozsah byl vypočten jako

$$RPM_{min} = 3200 * 0,9 = 2880 \text{ RPM}, \quad (29)$$

$$RPM_{max} = 3200 * 1,1 = 3520 \text{ RPM}, \quad (30)$$

Z naměřených základních frekvencí byla vypočítaná hodnota otáček ventilátoru 3188 RPM. Tato hodnota se nachází ve zvoleném intervalu a hodnota otáček za minutu odpovídá hodnotě určené výrobcem.



Obrázek 33 Frekvenční spektrum vibrací ventilátoru při napájecím napětí 12 V

Ze všech proměřených frekvenčních spekter pro jednotlivé velikosti napájecího napětí ventilátoru je rozpoznat, že vibrace v ose X dosahují celkově větší hodnoty amplitudy než vibrace v ose Y. Dominantní vibrace se tak pohybují ve směru rotační osy ventilátoru.

Z průběžných měření je vidět, že amplituda vibrací a dominance určitých frekvencí se mění. Z těchto důvodů by se pro diagnostiku generátoru vibrací měla použít hodnota pracovního napětí, kterou udává výrobce. Jedná se tak o hodnotu napájecího napětí, při které bude generátor pracovat a katalogový list od výrobce poskytuje parametry vztažené právě k této hodnotě.

## 7 ZADÁNÍ

1. Seznamte se s principem snímání akcelerace u předložených akcelerometrů a určete, jaké statické a dynamické parametry může akcelerometr změřit.
2. Pomocí mikropočítače Raspberry Pi a příslušné knihovny sestrojte program, který dokáže změřit a zobrazit náklon a rotaci desky, na které je umístěn tříosý digitální akcelerometr ADXL345 od firmy Analog Devices.
3. Pomocí vytvořeného programu změřte měřicí rozsah akcelerometru ADXL345. Proměřením daného rozsahu sestrojte korekční křivku a určete citlivost senzoru  $K$  v oblasti, kde absolutní chyba spadá do tolerančního pásma senzoru.
4. Změřte velikost a frekvenci vibrací, způsobenými otáčkami DC ventilátoru. Měření proveďte pomocí dvouosého akcelerometru ADXL203 od firmy Analog Devices.
5. Ze jištěných parametrů určete rychlost otáčení DC ventilátoru a rozhodněte, jaké frekvence jsou dominantní.



## 8 ZÁVĚR

Cílem práce bylo vytvořit laboratorní úlohu pro studenty VUT s MEMS akcelerometrem. Teoretická část práce zajišťuje seznámení s problematikou MEMS technologie a popisuje principy akcelerometrů.

Pro úlohu byly použity dva akcelerometry a to ADXL345, tříosý digitální akcelerometr a ADXL203, dvouosý analogový akcelerometr. Z důvodu, že ADXL345 je tříosý akcelerometr byl určen pro měření statických parametrů vzniklých působením gravitačního zrychlení země a díky digitálnímu výstupu zajišťuje uživatelsky přijatelnou komunikaci. Analogový akcelerometr ADXL203 byl zvolen pro měření dynamickou změnu hodnoty zrychlení, která je zajištěna umístěním na zdroj vibrací. V této úloze je zdrojem vibrací DC ventilátor od firmy Sunon. Důvodem zvolení analogového akcelerometru pro tuto část úlohy byl větší frekvenční rozsah zaznamenaných změn zrychlení oproti digitálním akcelerometrům.

Oba akcelerometry jsou umístěny na vytvořených deskách jejichž návrh je zobrazen v příloze A a B. Deska s ADXL345 obsahuje pouze pasivní prvky pro nastavení správné logické hodnoty při nevyužití komunikačních pinů, protože kontrolní elektronika pracující se snímačem je umístěna v pouzdře akcelerometru (digitální senzor). Deska s ADXL203 musela být doplněna o A/D převodník MCP3202 a level shifter pro zajištění správné komunikace.

K práci s akcelerometrem ADXL345 byl vytvořen aktuátor pro ruční změnu náklonu a rotace (změna zrychlení vůči gravitačnímu zrychlení pro všechny osy) a jeho části jsou zobrazeny v příloze C. Komunikace a zpracování naměřených hodnot je zajištěna připojením na piny určené pro komunikaci I<sup>2</sup>C mikropočítače Raspberry Pi 3b. Zde byla vytvořena knihovna v jazyce Python, která slouží studentům pro sestavení programu zajišťující práci s akcelerometrem pro část laboratorní úlohy.

Deska s akcelerometrem ADXL203 je umístěna na rámu ventilátoru, který slouží jako generátor vibrací. Časový průběh těchto vibrací je vzorkován pomocí změny zrychlení, kterou zaznamená akcelerometr. Takto vzorkovaný signál je pomocí analýzy FFT zpracován do frekvenčního spektra. Toto spektrum slouží studentům pro zaznamenání a vypočtení rychlosti otáčení lopatek ventilátoru a porovnání vibrací mezi sebou (osy X a Y akcelerometru). Záznam, zpracování a zobrazení dat zajišťuje napsaný program v jazyce Python.

Na vytvořených deskách a pomocí vytvořených programů bylo provedeno kontrolní měření. Pro měření náklonu a rotace bylo určené maximální možný měřený rozsah a rozsah měření, ve kterém hodnoty náleží tolerančnímu pásmu určeném teoretickými parametry, které poskytuje výrobce. Zde byla změřena také citlivost senzoru, která v měřicím rozsahu v tolerančním pásmu dosahuje hodnoty blízké 1. Tato hodnota naznačuje stejnou změnu měřeného vstupního a výstupního parametru. Z naměřených frekvencí ve frekvenčním spektru vibrací byla ze základní frekvence zjištěna rychlost otáčení v jednotkách RPM pro různá napájecí napětí DC ventilátoru. Z vyšších harmonických frekvencí je možné vidět nedokonalost výroby ventilátoru, který tak generuje vibrace obsahující více harmonických frekvencí.

# LITERATURA

- [1] An Introduction to MEMS (Micro-electromechanical Systems) [online]. Wolfson School of Mechanical and Manufacturing Engineering Loughborough University, Loughborough, Leics LE11 3TU: PRIME Faraday Partnership, 2002 [cit. 2019-11-08]. ISBN 1-84402-020-7. Dostupné z: [https://www.lboro.ac.uk/microsites/mechman/research/ipm-ktn/pdf/Technology\\_review/an-introduction-to-mems.pdf](https://www.lboro.ac.uk/microsites/mechman/research/ipm-ktn/pdf/Technology_review/an-introduction-to-mems.pdf)
- [2] BANDARI, Naghmeh. MEMS: The king of technology. Concordia University [online]. 2019 [cit. 2019-11-08]. Dostupné z: <https://www.concordia.ca/cunews/offices/vprgs/sgs/public-scholars-19/2019/05/28/mems-the-king-of-technology.html>
- [3] JAYAPANDIAN, J., K. PRABAKAR a O.K. SHEELA. Measure Laser Power With A Modified MEMS Pressure Sensor. Radio Locman [online]. 2013 [cit. 2019-11-08]. Dostupné z: <https://www.radiolocman.com/review/article.html?di=148185>
- [4] YEH, Po-Chen, Hao DUAN a Tien-Kan CHUNG. A Novel Three-Axial Magnetic-Piezoelectric MEMS AC Magnetic Field Sensor. Micromachines [online]. 2019, 10(10) [cit. 2019-11-08]. DOI: 10.3390/mi10100710. ISSN 2072-666X. Dostupné z: <https://www.mdpi.com/2072-666X/10/10/710>
- [5] D'ALESSANDRO, Antonino, Salvatore SCUDERO a Giovanni VITALE. A Review of the Capacitive MEMS for Seismology. Sensors. 2019, 19(14). DOI: 10.3390/s19143093. ISSN 1424-8220. Dostupné také z: <https://www.mdpi.com/1424-8220/19/14/3093>
- [6] ACAR, Cenk a Andrei SHKEL. MEMS Vibratory Gyroscopes: Structural Approaches to Improve Robustness. Springer Science+Business Media, LLC, 233 Spring Street, New York, NY 10013, USA: Springer, 2009. ISBN 978-0-387-09535-6.
- [7] Druhý Newtonův pohybový zákon - zákon síly. Encyklopedie fyziky [online]. 2019 [cit. 2019-11-14]. Dostupné z: <http://fyzika.jreichl.com/main.article/view/30-druhy-newtonuv-pohybovy-zakon-zakon-sily>
- [8] O'REILLY, Rob, Alex KHENKIN a Kieran HARNEY. Sonic Nirvana: Using MEMS Accelerometers as Acoustic Pickups in Musical Instruments. Analog Dialogue [online]. 2009, 43 [cit. 2019-11-14]. Dostupné z: <https://www.analog.com/en/analog-dialogue/articles/mems-accelerometers-as-acoustic-pickups.html>
- [9] Piezoelectric Accelerometers [online]. Metra Mess- und Frequenztechnik, 2001 [cit. 2019-11-09]. Dostupné z: <http://www.gracey.co.uk/downloads/accelerometers.pdf>
- [10] Design and Analysis of MEMS-Based Piezoresistive Accelerometer with Low Cross-Axis Sensitivity. IOSR Journal of Engineering. 2013, 3(4), 54-59. ISSN 2250-3021.
- [11] KAUR, Kalwinder. What is a Gyroscope? AZO Sensors [online]. 2013 [cit. 2019-11-28]. Dostupné z: <https://www.azosensors.com/article.aspx?ArticleID=252>
- [12] Gyroscope Sensor Working and Its Applications. Elprocus [online]. 2019 [cit. 2019-11-14]. Dostupné z: <https://www.elprocus.com/gyroscope-sensor/>
- [13] Coriolis force. Encyclopaedia Britannica [online]. 2019 [cit. 2019-11-21]. Dostupné z: <https://www.britannica.com/science/Coriolis-force>
- [14] Exploring the Application of Gyroscopes. CEVA [online]. 2019 [cit. 2019-11-21]. Dostupné z: <https://www.ceva-dsp.com/ourblog/exploring-the-application-of-gyroscopes/>

- [15] Analog Devices, Inc [online katalogový list]. ADXL345, 2015 [cit. 2019-11-21]. Dostupné z: <https://www.analog.com/media/en/technical-documentation/data-sheets/ADXL345.pdf>
- [16] Analog Devices, Inc [online katalogový list]. ADXL203, 2015 [cit. 2019-11-28]. Dostupné z: [https://www.analog.com/media/en/technical-documentation/data-sheets/adxl103\\_203.pdf](https://www.analog.com/media/en/technical-documentation/data-sheets/adxl103_203.pdf)
- [17] Microchip Technology Inc. [online katalogový list]. MCP3202, 2011 [cit. 2019-11-21]. Dostupné z: <http://ww1.microchip.com/downloads/en/DeviceDoc/21034F.pdf>
- [18] What is a Raspberry Pi? Raspberry Pi [online]. [cit. 2019-11-22]. Dostupné z: <https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/>
- [19] Raspberry Pi 3 Model B. Raspberry pi [online]. [cit. 2019-12-04]. Dostupné z: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>
- [20] Raspberry Pi 3 Model B. Terraelectronica [online]. [cit. 2019-12-04]. Dostupné z: [https://www.terraelectronica.ru/pdf/show?pdf\\_file=%252Fds%252Fpdf%252FT%252FTec%252FhicRP3.pdf](https://www.terraelectronica.ru/pdf/show?pdf_file=%252Fds%252Fpdf%252FT%252FTec%252FhicRP3.pdf)
- [21] Raspberry Pi Documentation: GPIO [online]. [cit. 2020-02-19]. Dostupné z: <https://www.raspberrypi.org/documentation/usage/gpio/>
- [22] Apt-get command in Linux with Examples. GeeksforGeeks [online]. [cit. 2020-02-29]. Dostupné z: <https://www.geeksforgeeks.org/apt-get-command-in-linux-with-examples/>
- [23] Python Package Index [online]. [cit. 2020-02-29]. Dostupné z: <https://pypi.org/>
- [24] Raspberry Pi Documentation: Installing Python packages [online]. [cit. 2020-02-29]. Dostupné z: <https://www.raspberrypi.org/documentation/linux/software/python.md>
- [25] Raspberry Pi SPI and I2C Tutorial [online]. [cit. 2020-02-29]. Dostupné z: <https://learn.sparkfun.com/tutorials/raspberry-pi-spi-and-i2c-tutorial/all>
- [26] PEDLEY, Mark. Tilt Sensing Using a Three-Axis Accelerometer. Freescale Semiconductor [online]. 2013, rev. 6 [cit. 2019-11-28]. Dostupné z: <https://www.nxp.com/docs/en/application-note/AN3461.pdf>
- [27] NXP Semiconductors N.V. I2C - bus specification and user manual [online]. In: . 2014, s. 64 [cit. 2020-03-18]. Dostupné z: <https://www.nxp.com/docs/en/user-guide/UM10204.pdf>
- [28] LINUX KERNEL ORGANIZATION, INC. SMBus Protocol Summary [online]. [cit. 2020-03-18]. Dostupné z: <https://www.kernel.org/doc/Documentation/i2c/smbus-protocol>
- [29] Python GUI – tkinter. GeeksforGeeks [online]. [cit. 2020-03-20]. Dostupné z: <https://www.geeksforgeeks.org/python-gui-tkinter/>
- [30] ROSENTHAL, David a Sun MICROSYSTEMS, MARKS, Stuart W. a SUNSOFT, INC., ed. Inter-Client Communication Conventions Manual [online]. [cit. 2020-03-20]. Dostupné z: <https://tronche.com/gui/x/icccm/sec-4.html#s-4.2.8>
- [31] Measuring Vibration with Accelerometers. National Instruments [online]. 2019 [cit. 2019-11-30]. Dostupné z: <https://www.ni.com/cs-cz/innovations/white-papers/06/measuring-vibration-with-accelerometers.html>
- [32] BILOŠ, Jan a Alena BILOŠOVÁ. Aplikovaný mechanik jako součást týmů konstruktérů a vývojářů: studijní opora. Ostrava: Vysoká škola báňská - Technická univerzita Ostrava, Fakulta strojní, 2012. ISBN 978-80-248-2755-1.
- [33] PyQtGraph: Scientific Graphics and GUI Library for Python [online]. [cit. 2020-03-27]. Dostupné z: <http://www.pyqtgraph.org/>
- [34] About Qt. Qt wiki [online]. 2019 [cit. 2020-03-27]. Dostupné z: [https://wiki.qt.io/About\\_Qt](https://wiki.qt.io/About_Qt)

- [35] System-specific parameters and functions. Python [online]. [cit. 2020-03-27]. Dostupné z: <https://docs.python.org/3/library/sys.html>
- [36] Python Programming/Interactive mode. Wikibooks: Open books for an open world [online]. [cit. 2020-03-28]. Dostupné z: [https://en.wikibooks.org/wiki/Python\\_Programming/Interactive\\_mode](https://en.wikibooks.org/wiki/Python_Programming/Interactive_mode)
- [37] Python hasattr(). Programiz: Learn to Code for Free [online]. Parewa Labs Pvt. [cit. 2020-03-28]. Dostupné z: <https://www.programiz.com/python-programming/methods/built-in/hasattr>
- [38] NumPy [online]. NumPy developers, 2020 [cit. 2020-03-16]. Dostupné z: <https://numpy.org/>
- [39] SciPy library. SciPy.org [online]. SciPy developers, 2020 [cit. 2020-03-16]. Dostupné z: <https://www.scipy.org/scipylib/index.html>
- [40] HORÁK, David. DISKRÉTNÍ TRANSFORMACE [online]. Rychvald, 2012 [cit. 2020-04-08]. Dostupné z: [http://mi21.vsb.cz/sites/mi21.vsb.cz/files/unit/diskretni\\_transformace.pdf](http://mi21.vsb.cz/sites/mi21.vsb.cz/files/unit/diskretni_transformace.pdf). Projekt. Vysoká škola báňská – Technická univerzita Ostrava a Západočeská univerzita v Plzni.
- [41] SMÉKAL, Zdenek. Analýza signálu a soustav - BASS. Brno, 2012. Skripta. FEKT Vysokého učení technického v Brně.
- [42] SUNONWEALTH ELECTRIC MACHINE INDUSTRY CO. [online katalogový list]. EE80251B1-000U-A99, 2014 [cit. 2020-04-28]. Dostupné z: [http://portal.sunon.com.tw/pls/portal/sunonap.sunon\\_html\\_d\\_pkg.open\\_file?input\\_file\\_name=7264646F632F3230313430312F3137363538332F28443038303338313230472D3030292D342E706466](http://portal.sunon.com.tw/pls/portal/sunonap.sunon_html_d_pkg.open_file?input_file_name=7264646F632F3230313430312F3137363538332F28443038303338313230472D3030292D342E706466)

# SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

SPI	sériové periferní rozhraní (Serial Peripheral Interface), druh komunikace
I <sup>2</sup> C	multi-masterová počítačová sériová sběrnice (Inter-Integrated Circuit)
A/D	převodník z analogového signálu na digitální
D/A	převodník z digitálního signálu na analogový
FIFO	druh paměti typu první dovnitř, první ven (First In First Out)
HDTV	formát vysílání televizního signálu s velkým rozlišením
USB	univerzální sériová sběrnice, moderní způsob připojení periférií
LAN	lokální síť, počítačová síť pokrývající malé území (domácnost)
LGA	druh pouzdra, ve kterém je umístěn obvod (Land Grid Array)
LSB	bit binárního čísla značící nejmenší mocninu (Least Significant Bit)
LCC	druh pouzdra, ve kterém je umístěn obvod (Leadless Chip Carrier)
RPM	počet otáček za minutu (Revolutions Per Minute)
DC	stejnoseměrný elektrický proud (Direct Current)
FET	druh unipolárního tranzistoru (Field-Effect Transistor)
FFT	rychlá Fourierova transformace (Fast Fourier Transform), druh algoritmu
DFT	diskrétní Fourierova transformace (Discrete Fourier Transform)

# SEZNAM OBRÁZKŮ

Obrázek 1 Vytvořené MEMS struktury pomocí mikroobráběcích procesů [2].....	2
Obrázek 2 Princip piezorezistivní MEMS snímače [3] .....	3
Obrázek 3 Princip piezoelektrického MEMS snímače [3] .....	3
Obrázek 4 Princip kapacitního senzoru s proměnou mezerou (vlevo) a proměnou plochou (vpravo) [5] .....	4
Obrázek 5 Princip kapacitního akcelerometru s proměnou mezerou [8].....	6
Obrázek 6 Princip MEMS gyroskopu [14] .....	7
Obrázek 7 Popis a adresy digitálního akcelerometru ADXL345 [15].....	8
Obrázek 8 Vnitřní blokové schéma digitálního akcelerometru ADXL345 [15] .....	9
Obrázek 9 Vnitřní blokové schéma analogového akcelerometru ADXL203 [16] .....	10
Obrázek 10 Vnitřní blokové schéma A/D převodníku MCP3202 [17] .....	11
Obrázek 11 Rozložení součástek a konektorů na desce Raspberry Pi 3 model B [20] ..	12

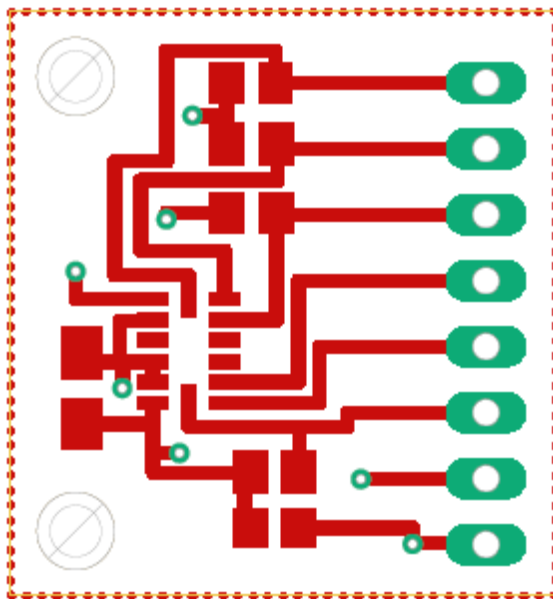
Obrázek 12 Zobrazení funkčnosti pinů mikropočítače Raspberry Pi 3b [21] .....	13
Obrázek 13 Povolení sériové komunikace SPI a I2C pomocí terminálu na mikropočítači Raspberry Pi.....	15
Obrázek 14 Určení směru jednotlivých os u tříosého akcelerometru [15] .....	17
Obrázek 15 Blokové schéma principu úlohy pro měření náklonu a rotace .....	17
Obrázek 16 Závislost zvolené hodnoty pull-up rezistoru na a) kapacitě sběrnice b) napájecího napětí sběrnice .....	18
Obrázek 17 Schéma zapojení desky s akcelerometrem ADXL345 .....	19
Obrázek 18 Schéma rotačního stolku RL60-A .....	21
Obrázek 19 Vytvořený aktuátor pro změnu náklonu a rotace .....	21
Obrázek 20 Vytvořené okno pro zobrazení naměřených hodnot .....	25
Obrázek 21 Blokové schéma úlohy pro měření vibrací.....	29
Obrázek 22 Schéma zapojení level shifteru pro převod 5 V sběrnice na 3,3 V sběrnici	31
Obrázek 23 Upevnění desky s akcelerometrem ADXL203 na rámu DC ventilátoru.....	32
Obrázek 24 Vytvořené okno pro zobrazení grafů s naměřenými hodnotami .....	35
Obrázek 25 Graf závislosti nastavené hodnoty rotace na změřené hodnotě.....	40
Obrázek 26 Graf korekční křivky pro měření rotace .....	41
Obrázek 27 Graf závislosti nastavené hodnoty náklonu na změřené hodnotě .....	42
Obrázek 28 Graf korekční křivky pro měření náklonu .....	42
Obrázek 29 Frekvenční spektrum vibrací ventilátoru při napájecím napětí 4,5 V .....	44
Obrázek 30 Frekvenční spektrum vibrací ventilátoru při napájecím napětí 6 V .....	44
Obrázek 31 Frekvenční spektrum vibrací ventilátoru při napájecím napětí 8 V .....	45
Obrázek 32 Frekvenční spektrum vibrací ventilátoru při napájecím napětí 10 V .....	45
Obrázek 33 Frekvenční spektrum vibrací ventilátoru při napájecím napětí 12 V .....	46

## SEZNAM TABULEK

Tabulka 1 značení kabelů desky s akcelerometrem ADXL203.....	31
Tabulka 2: Porovnání rychlosti mezi algoritmem DFT a FFT .....	36
Tabulka 3 Naměřené hodnoty úhlů rotace a náklonu akcelerometrem ADXL345 .....	39
Tabulka 4 Naměřené hodnoty frekvenčního spektra pro měřené hodnoty napětí .....	43

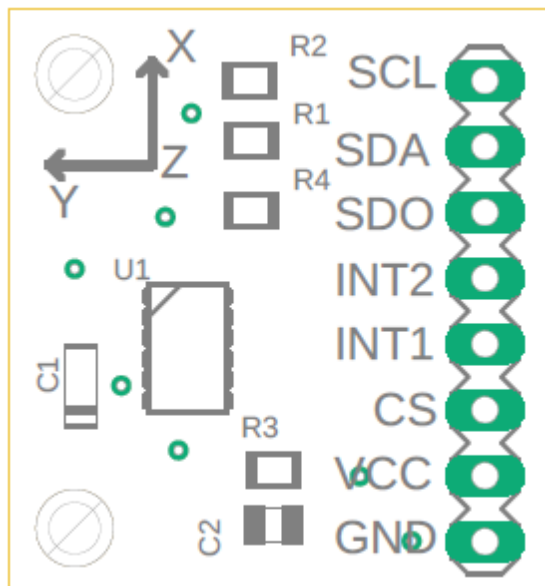
## A NÁVRH DESKY S AKCELEROMETREM ADXL345

### A.1 Deska plošného spoje s ADXL345 – Top (strana součástek)



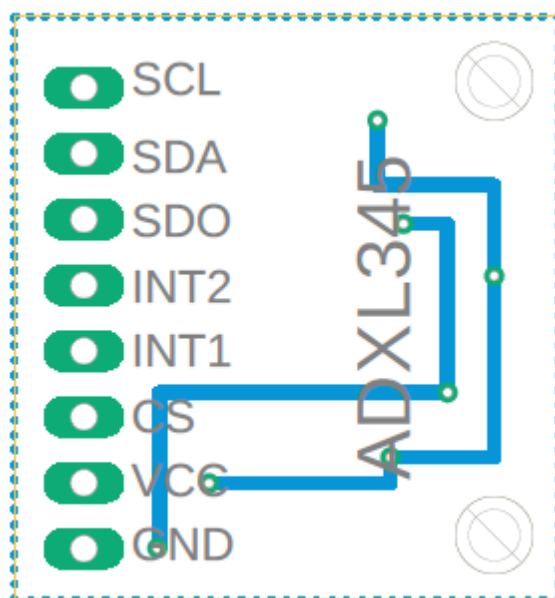
Rozměry 21 x 22,5 mm, měřítko M1:3

## A.2 Deska plošného spoje s ADXL345 – rozmístění součástek



Rozměry 21 x 22,5 mm, měřítko M1:3

## A.3 Deska plošného spoje s ADXL345 – Bottom (zrcadleno)

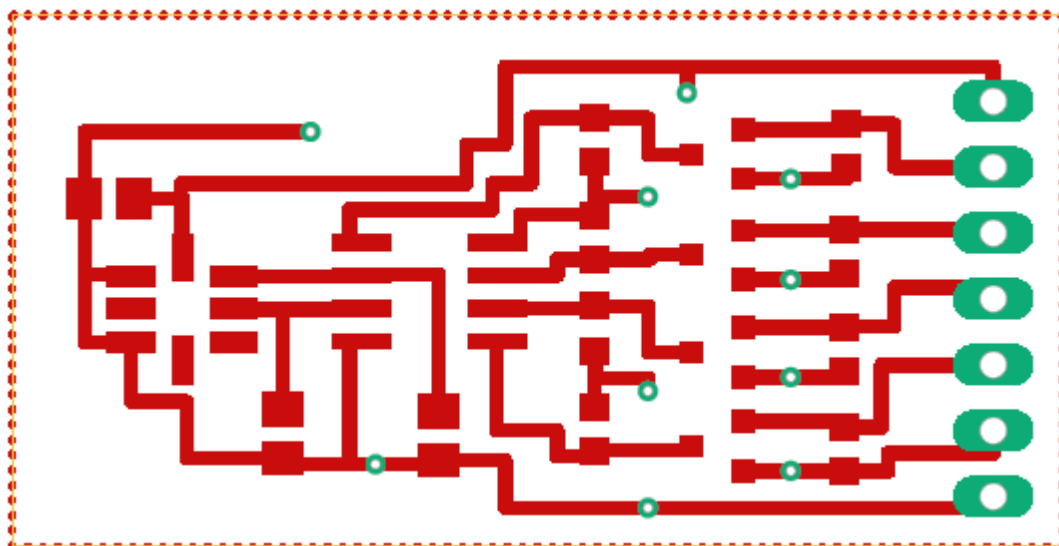


Rozměry 21 x 22,5 mm, měřítko M1:3



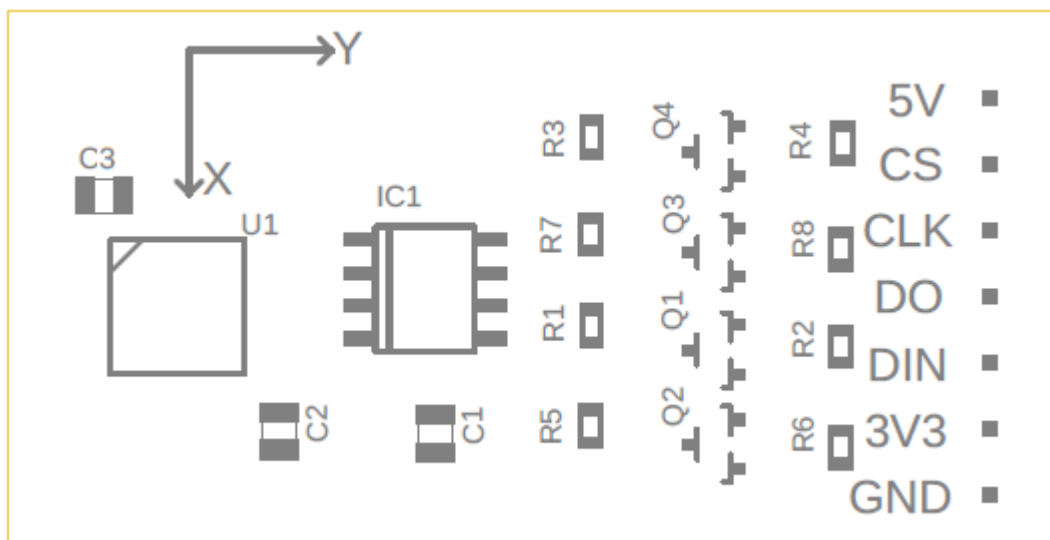
## B NÁVRH DESKY S AKCELEROMETREM ADXL203

### B.1 Deska plošného spoje s ADXL203 – Top (strana součástek)



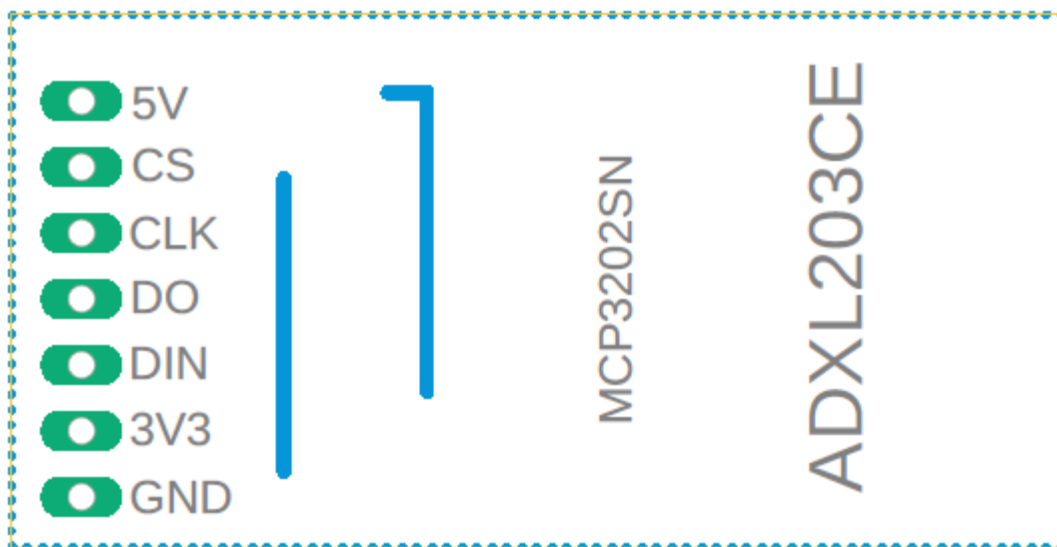
Rozměry 40,5 x 20,5 mm, měřítko M1:3

### B.2 Deska plošného spoje s ADXL203 – rozmístění součástek



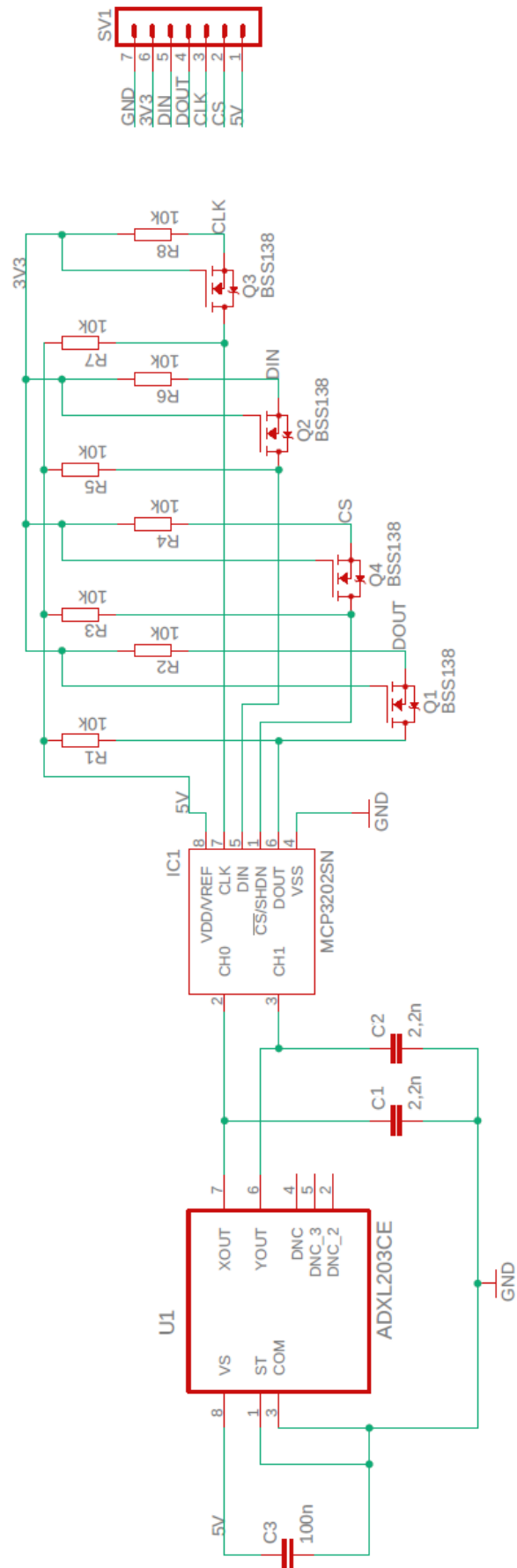
Rozměry 40,5 x 20,5 mm, měřítko M1:3

### B.3 Deska plošného spoje s ADXL203 – Bottom (zrcadleno)



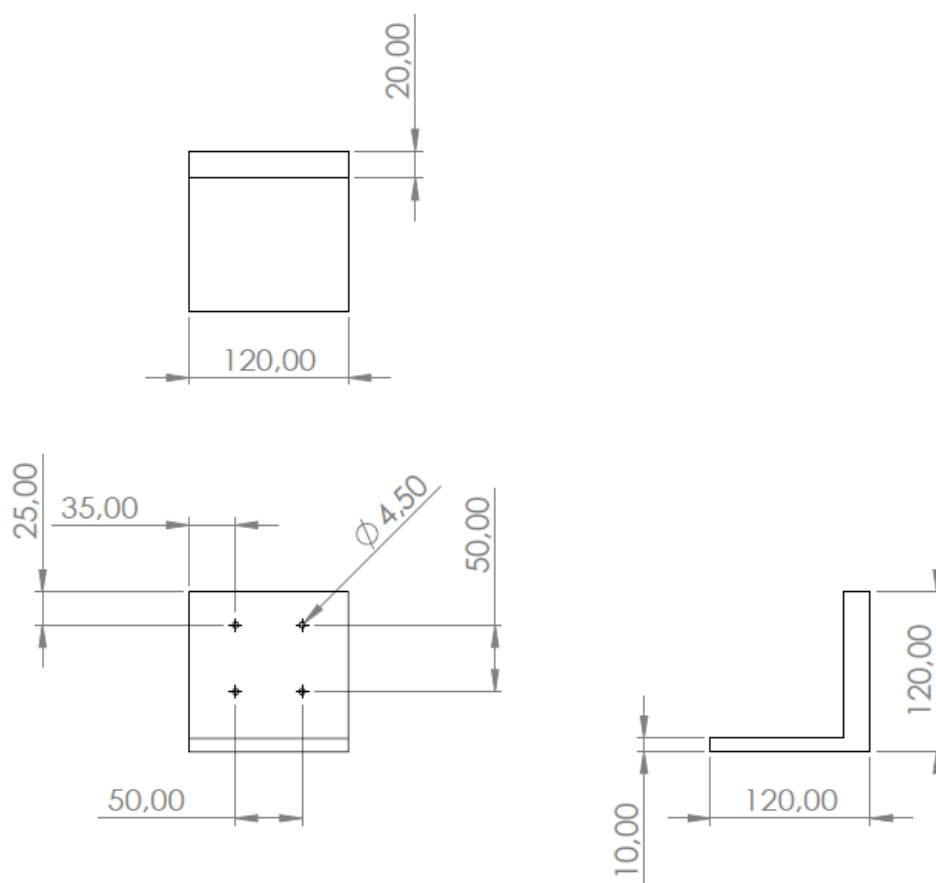
Rozměry 40,5 x 20,5 mm, měřítko M1:3

### B.4 Schéma zapojení desky s ADXL203

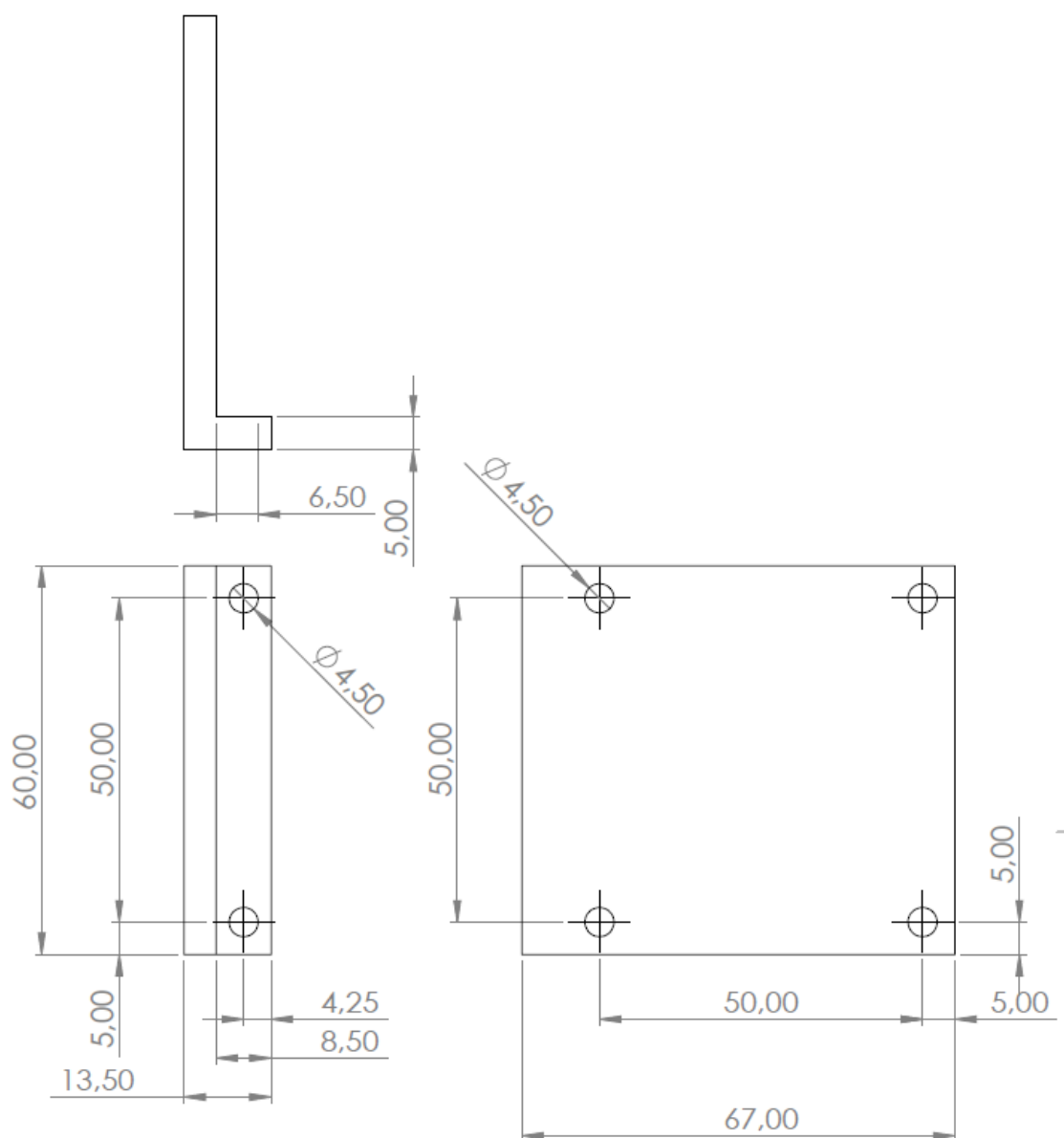


## C NÁVRH ČÁSTÍ AKTUÁTORU PRO ZMĚNU NÁKLONU A ROTACE

### C.1 Návrh základny aktuátoru (měřítko M1:5)



## C.2 Návrh držáku pro rotační stůlek 2 (měřítko M1:1)



### C.3 Návrh držáku pro desku s akcelerometrem ADXL345 (měřítko M1:1)

